QL Today

Volume 7
Issue 3
Sept./October
2002

ISSN 1432-5454

The Magazine about QL, QDOS, Sinclair Computers, SMSQ...

QL Logo

Will we ever get something like the ATARI symbol or the Mac apple?

QPC Hints & Tips

Q 60 Review

All the series continue: Assembler EasyPTR QPTR C

Contents

Advertisers

in alphabetical order

QL Today

ISSN 1432-5454

German office & Publisher:

Jochen Merz Software
Im stillen Winkel 12
47169 Duisburg
Germany

Tel. +49 203 502011
Fax +49 203 502012
Box1 +49 203 502013
Box2 +49 203 502014
Mobile +49 1701 222277
email: JMerz@j-m-s.com
email: QLToday@j-m-s.com

English office:

Q Branch
20 Locks Hill
Portslade
BN41 2LB
United Kingdom

Tel. +44 1273 386030
Mobile +44 7836 745501
Fax +44 1273 381577
email: qbranch@qbranch.demon.co.uk
email: QLToday@j-m-s.com

Editor:

Dilwyn Jones
41 Bro Emrys
Tal-Y-Bont, Bangor
Gwynedd
United Kingdom LL57 3YT

Tel. +44 1248 354023
Fax +44 1248 354023
email: dilwyn.jones@dj.softnet.co.uk
email: QLToday@J-M-S.com

Co-Editor:

Bruce Nicholls
38 Derham Gardens
Upminster
Essex RM14 3HA
United Kingdom

Tel +44 1708 510764
Fax +44 870 0568755
email: qltoday@q-v-d.demon.co.uk
email: QLToday@j-m-s.com

QL Today is published bi-monthly, our volume begins on beginning of June. Subscriptions begin with the current issue at the time of sign up. Please contact the German or English office for current subscription rates.

We welcome your comments, suggestions and articles. YOU make **QL Today** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email or into one of the JMS-BBS's. We prefer ASCII, Quill or text87 format. Pictures may be in _SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

Article and Advertising deadlines are as follows:

Issue 1: 30 April	Issue 2: 30 June
Issue 3: 30 August	Issue 4: 30 October
Issue 5: 30 December	Issue 6: 28 February

Editorial

Dilwyn Jones

Another issue of QL Today, another cover disk. This time we bring you a disk with Thierry Godefroy's Atapi/CD software and several other programs associated with reading QXL.WIN media on systems such as Qubide v2 and Q40/Q60 systems, which normally use non-QXL.WIN media. Having supplied these and if they prove successful and popular we may, just may, at some point in the future be able to bring you the first ever purely QL cover CD!

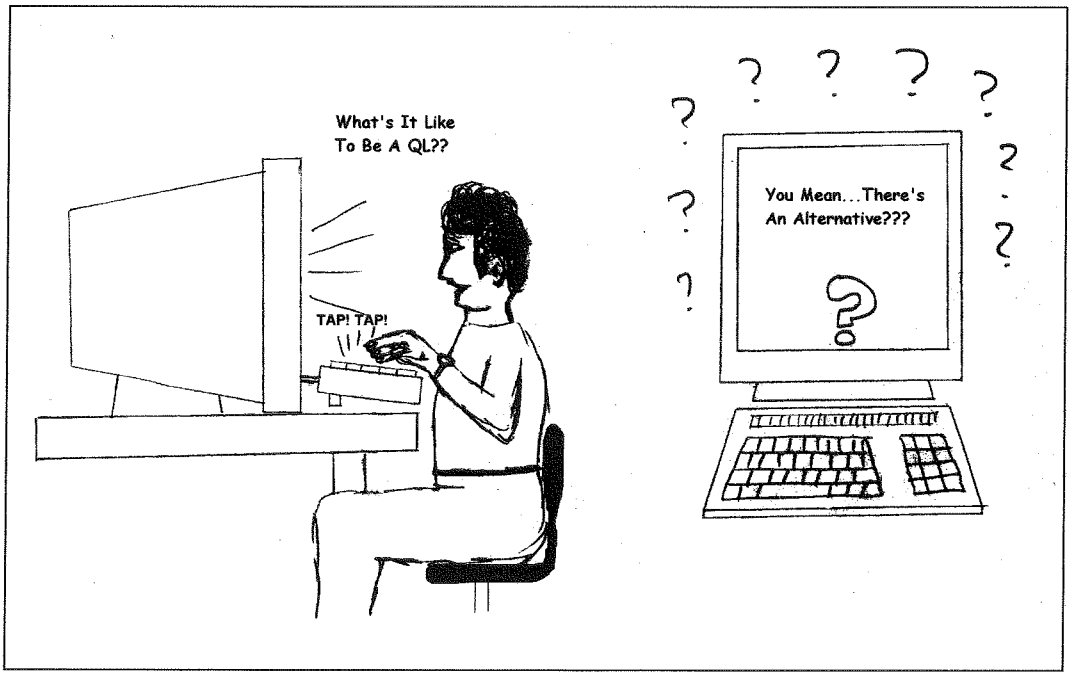
Wolfgang Uhlig would like us to mention that we split his article in two parts, that it was not his intention. Well, the situation was that a short time before publication we had very little material to hand, we asked around for articles in panic mode, and got a lot - which of course then just had to be split in order to turn nobody down since several people had gone out of their way to help us.

And as we are fairly short for this issue too, can we please issue a little plea for a few extra articles for the next issue? Things have gone very quiet as usual over the summer period, so it would be nice to receive a lot more for the next few issues!

Jochen Merz has asked me to mention that one of the phone numbers for his BBS (Bulletin Board System) (the 502013) should not be used anymore - only the 502014 number should be used!

I picked up an interesting snippet on the ql-users mailing lists recently when Jon Dent let Slip (in-joke, sorry) that his soql TCP/IP system was now very close to having PPP.I'm wetting your appetite with all this terminology...could we be fairly close to the holy grail of internet and email from a QL at last? And there's quite a few interesting projects going on in the background which could generate a fair bit of interest later in the year. Some of them are projects which have been ongoing for a while, others quite new, so keep reading QL Today for all the news!

Cartoon



NEWS

QL Kick ROM for UAE

by Mark Swift

After months of neglect, I have updated my web site with a couple of new files.
The first file is a Kick ROM for UAE that boots as a QL.
UAE is an amiga emulator that runs on most platforms. Normally you have to have a real Amiga to use this as the emulator requires a real Amiga ROM file (Kick ROM).
The Kick ROM file on my web page is a replacement ROM that boots UAE as a QL instead (QDOS Classic). i.e. it doesn't require an Amiga Kick ROM.
You can download the Kick ROM for UAE at:
http://pages.unisonfree.net/mswift/ql/files/QZ/CLSC/RUN/UAE/UAEQLKickROM.zip

The second file is a RAW QDOS disk image for use with the UAE QL Kick ROM.
This can be downloaded at:
http://pages.unisonfree.net/mswift/ql/files/QZ/CLSC/RUN/UAE/UAEQLADF.zip

I have tested the QL Kick ROM on MaxUAE (Mac OS X) and WinUAE (PC) and it works OK. Supports sound and RAW MFM disk images – but otherwise is currently a bit limited.

Just Words! has moved!

Geoff Wicks writes:
Please note the new contact details for Just Words!
Address:
Geoff Wicks,
56 Peveril Crescent,
West Hallam,
Derbyshire,
DE7 6ND.
Telephone: **+44 (0)115 930 3713**
email: **geoffwicks@hotmail.com**
Web: **http://members.lycos.co.uk/geoffwicks/justwords.htm**

Wordscheck Update

Dilwyn Jones

My Wordscheck program is now at version 1.9, which includes a change to the method of calculating free memory to improve the way it works on the current versions of QPC2.
The program can be downloaded from the My Freeware page on
www.soft.net.uk/dj/software/freeware/freeware.html
and will be added to the PD library and CD-ROMs on which it appears.

QL CD-ROMS in Norh America

The Dilwyn Jones / Q-Celt Computing range of QL CD-ROMs is now available in North America from:
Phoebus R. Dokos,
941 Lilac Street Apt.#1
Indiana
PA 15701-3340
U.S.A.
Tel: **+1(724) 464 0199**
Please contact Phoebus direct for details of prices and payment methods.
These CD-ROMs can be read direct from QPC, QemuLator v2, QXL, uQLx and from Q40/Q60 and Qubide v2 with CD-ROM drives and Thierry Godefroy's excellent Atapi/CD extensions.

TCP/IP... getting there!

An interesting announcement cropped up a few weeks ago on the ql-users mailing list. Michael Grunditz first wrote:
"I have compiled ulP on my Q40 , and on uqlx , but I dont know if it fully works yet. I hope I get some more time this week to build the slip network needed. So, soon you might be surfing to a webserver hosted on smsq/e!"
Then, up popped an interesting reply from Jon Dent, who is still grafting away on his soql system for QDOS:
"This looks very interesting. An alternative to SOQL. Also uses SLIP I see. P.S. Don't tell anyone but I think I'm close to a working PPP on SOQL"
So – the holy grail of internet and email access on a QL might not be too far away at last!

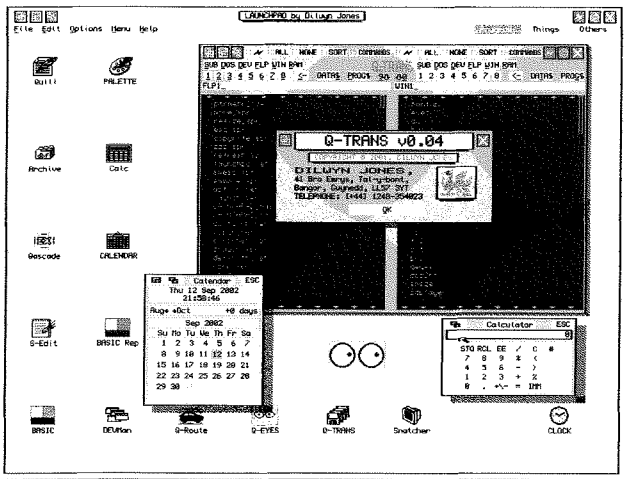
QL CD-ROMs

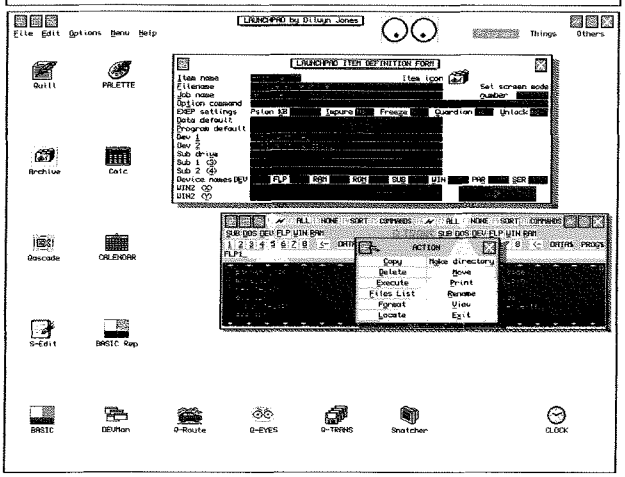
The QL Emulators CD is now at v1.21. By popular request I went back to including a copy of QPCv1 for DOS users. It also includes a number of lesser updates to the range of free software bundled on the CD.
The DJ Bargain Software Bundle is at version 1.8, including all of my latest freeware software.
The QL Documentation CD is now at release 2 and includes a huge number of added QL documentation.
The biggest update is to the PD Library CD. At the

time of writing the general PD disks section alone was up to 82 disks with a large range of programs still awaiting classification.

The Literature CD mentioned in the last issue should be available by the time you read this. Mike Edwards has obviously put a lot of work into this and it should be available at a price of £10.00

The Q-Trans pointer driven file transfer program mentioned in the news pages of Volume 7 Issue 1 of QL Today has been significantly enhanced since then and is now in its fourth release incorporating a number of features added at the request of early users. If you are using a version earlier than Beta 0.03 it is well worth upgrading, both for the bug fixes and added facilities, including a new file finder command (search for files either by filename or text content), ability to execute non-job files via File Info 2 or view files via File Info 2, integration of more commands into the context menus, which can also be called up via direct commands rather than context menus and the option to specify left and right files window drive/directory via an option command parameter at startup to override the configured defaults. Many of these changes are to allow it to be integrated more seamlessly with the Launchpad QL desktop program which is also being slowly but surely pushed forward - more news on that soon.





QCDEZE

by Duncan Neithercut

The latest release of QCDEZE is v1.04 and associated files in a zip archive. The new version of QCDEZE has some bug fixes and some additional features.

The major additional features are: now seamlessly browses QXL.win files on CD roms and also browses other devices like hard disks. Can also run on QPC2 but only in mode 4 icons. An additional associated file with this release is an early version of a file converter which can display on Q40/Q60 high colour screens old QL mode 4 and 8 pic files as well as Q40 pic files, Amiga IFF files, BMPs and PCXs. There are other minor additions explained in the documents.

Half a Million Words from RWAP Software!

We have now released a 564,000+ word dictionary for use with QTyp.

This dictionary comes compressed and still needs over 1Mb of memory to work, so a Super Gold Card is a minimum requirement.

We can supply it on HD disk for £10 or on CD for £15 - the CD version includes the dictionary as an ASCII word list (nearly 6Mb in size).

The dictionary has been compiled by Paul Meridinian, who compiled the Mega Dictionary for Digital Precision's Spellchecker. We are looking into making a version available for that program if enough users request it.

RWAP Software
7 Common Road,
Kinsley,
Pontefract,
West Yorkshire,
WF9 5JR
TEL: **01977 614299**
http://hometown.aol.co.uk/rwapsoftware

QemuLator News

from Daniele Terdina

A few months after the old web site at geocities was discontinued, I managed to move it to a new address:

http://users.infoconex.com/daniele/q-emulator.html

Registered users of the Windows version can download an update from:

http://users.infoconex.com/daniele/winql.html

Cover Disk for Volume 7 Issue 3

This cover disk contains a batch of software which will prove to be very useful for those handling QXL.WIN file container systems on certain systems such as Qubide v2 and Q40/Q60, as one of the things greatly simplified by these programs is the reading of media in QXL.WIN format such as many of the available QL CD-ROMs. The Q40/Q60 seems to use Atari-style hard disk systems, while the Qubide system is based on the older Rebel hard disk system.

The Atapi & CD extensions software by Thierry Godefroy is the key to all this, most of the other programs rely on having it present. Please read the accompanying documentation files for further details.

We hope that having brought you this cover floppy disk initially, we MAY at some point in the future be able to innovate further by providing the first ever truly QL format CD-ROM!

The programs are supplied zipped to get them to fit on a 720KB disk. Simply LRUN the boot program on the disk, select which package to unzip, have a few blank formatted disks to hand and wait for UNZIP to do its work. The full official Infozip QDOS/SMSQE unzip and zip packages are maintained by Jonathan Hudson and available for download via his website on

http://www.daria.co.uk

The four packages supplied are:

Atapi and CD-ROM system extensions by Thierry Godefroy, needed to run the other packages on this cover disk.

QXLWIN copier for Q40/Q60 by Wolfgang Lenerz. Allows you to copy files from a QXL.WIN file held on CD-ROM on a Q40/Q60 using Thierry Godefroy's CD-ROM driver extensions above.

Qwirc by Per Witte. QL Winchester Information And Rename Console.

QCDEZE. Duncan Neithercutt's GUI front end which enables browsing of any ISO-9660 formatted CD-ROM from a Q40 or other QL system compatible with Thierry Godefroy's ATAPI CD_ROM drivers (above) when these are installed. Version 1.04 now includes a useful graphics conversion utility, as well as a text file with information about the ISO-9660 CD-ROM formats.

Replacement Manuals

Dilwyn Jones

There is still a healthy trade in second hand QL hardware judging by emails sent to the QL mailing lists and hearing from traders like Rich Mellor. One possible problem, though, is obtaining replacement manuals for many bits of QL hardware. A browse through a "bring and buy" stall at a Quanta workshop for example will show that few items sold second hand will come with their manuals. This can be a real problem, as many of the original QL traders have ceased selling QL products. Indeed, I get a number of requests asking if I have spare copies of older manuals.
Fortunately, many have allowed distribution of their manuals as replacement manuals and this is probably because it saves them having to reply to queries I suppose.
Several QL hardware traders both former and current have given me permission to supply replacement manuals on a non-profit basis (essentially PD-library type terms) and some of these are available from my website. The manuals for Miracle Systems products such as Trump/Gold Cards, QXL and ED drives are available in electronic form from me, as are manuals for several of the TF Services hardware products (Minerva currently excepted) and even the manual for the QL itself. Some, where space permits, are available on my website

http://www.soft.net.uk/dj/index.html

or the north American mirror on

www.dokos-gr.net/~dj/

or via my PD library service, or on a CD-ROM called the QL Documentation CD. It's been a lot of hard work getting all these together, and thanks are due to a lot of people such as Tim Swenson, Dave Westbury and Tony Firshman who helped me with the project by supplying manuals on disk, but at least it's done now.
Any more old hardware manuals anyone cares to scan and OCR gratefully received as long as it can be legally used!

E-Mail - still a fast communication method?

Jochen Merz

I am actually wondering. I hope this article does not sound too negative, but I am writing this in a rather negative mood. I think the trigger was an E-mail which I received from a customer this morning, asking if I could be helpful in getting in email contact with other QL companies, as he's trying but without any success - no reply at all. His last sentence was "What happens to QLers???". That was not the first time customers asked for help. Well, I am having the same problem and asked around if somebody else knew if there are E-mail problems with certain people. I also noticed this several times being asked in the newsgroups. Tony Firshman often seems to have the same problem. I am sure others are having similar experiences and this is not very good.

If you think filling one page is rather a lot, then you may be right. However, the current E-Mail situation is not only very annoying and unnecesserily unprofessional, it shows that the "personal" bit, the difference between the QL scene and "the rest" seems to have vanished. I find this very worrying.

It seems that the QL world splits in two halfs - E-mail wise: there are those who reply immediately to E-mails all the time, and there are those where it takes weeks and reminder E-mail after reminder E-mail to get a single thing sorted out. And this is where I start to wonder why. It seemed to have got worse over the last year or so. Is the E-mail "hype" gone? Is it a matter of too much spam (I'm sure we're all suffering)? Is it lazyness? Do all the E-mails get lost? Or are QLers not worth the effort anymore? Hard to tell, and probably different in every case. But what effect does it have to the writer of the E-mail? That's important, and that's usually negative if nothing at all happens. Which impression is given if a customer does not get a reply for weeks, but the person addressed replies happily to newsgroups etc.

I am always trying to reply to every E-mail within 48 hours, and usually reply even within 24 hours. From my experience and the way people write to me and react, and from what can be read about "internet politeness rules", 48 hours is the absolute maximum. What's so difficult about a quick reply "I am very busy right now but it should cool down in three days" or something like that?

It may be a different matter on personal E-mails, but I think for a business it is absolutely essential to be quick on E-mail replies, otherwise it is pretty pointless. Customers usually write because they have an urgent request, and don't expect a reply 3 weeks later if they are lucky. I am sure everybody agrees - that's very unprofessional and not the way it should be.

In my main business life (outside the QL scene), I have to deal with companies which don't bother to reply to E-mails. Well, if I haven't heared anything after three mails, this company is history for me. I wouldn't really want to see that happen in the QL scene - although it is actually happening already: I place orders for goods which a customer has placed, and I don't get replies. It requires two, three additional mails to get a single reply - but the problem is: I seem to be unreliable in my customer's view... and that's something I really hate. What other option do I have: tell the customer I can't get it for him 'cause the company is too unreliable? What sort of view would this give about the QL scene in general?

I start to think that, when I have to write five E-mails to eventually get a reply or no reply at all, and I have to call the person anyway to get something done, why shall I write all the E-mails in the first place and waste my time? The effort to repeatedly send E-mails, wait, check again etc. seems to slow down communication (and, worst, slow down about whatever activity the E-mail was about) very much!

I don't want to list bad/good examples here. If you're one of the E-mail "collectors", who don't reply for whatever reason, then please consider that your reason might not match the impression the writer of the E-mail will have. If you are busy because of whatever reason, let at least the writer know. A quick reply takes 30 seconds, and if you are getting too many mails and you are very busy for a while, setup an auto-responder which replies ... this takes 5 minutes once and the auto-responder will do the work for you. But please, don't ignore the mails or wait until the next three reminder mails arrive. Remember, there's a person on the other end of the E-mail "conversation" waiting for your reply ...

Gee Graphics! (on the QL?) - Part 30

H. L. Schaaf

"And Now, Least Connection of all the dots".....

This time we add to the menu the Minimum Spanning Tree, another subset of the Delaunay triangulation. For all the paths that connect all the dots, there is a minimal path length. A path with that minimal length is a Minimum Spanning Tree or MST for short (pun intended). There may be more than one such path.
I've adapted Algorithm 422 from the Communications of the ACM, April 1972. It uses the technique suggested by Dijkstra in his paper, "A note on two problems in connection with graphs" Numerische Mathematik Vol. 1 No. 5 pp 269-271 October 1959
Here are some excerpts from Dijkstra's recent obituary:

"" from the New York Times, August 10, 2002, By JOHN MARKOFF

Edsger Wybe Dijkstra, whose contributions to the mathematical logic that underlies computer programs and operating systems make him one of the intellectual giants of the field, died on Monday at his home in Nuenen, the Netherlands. He was 72.
Dr. Dijkstra is best known for his shortest-path algorithm, a method for finding the most direct route on a graph or map
The shortest-path algorithm, which is now widely used in global positioning systems and travel planning, came to him one morning in 1956 as he sat sipping coffee on the terrace of an Amsterdam cafe.
It took him three years to publish the method, which is now known simply as Dijkstra's algorithm. At the time, he said, algorithms were hardly considered a scientific topic.
Dr. Dijkstra, an advocate of an approach known as structured programming, wrote a short research note in the March 1968 edition of the journal Communications of the ACM that became legendary. Titled "The GO TO Considered Harmful," it argued against the complexity of a feature in programming languages like Fortran and Basic that permitted programmers to write convoluted programs that jump around haphazardly.

....in the early 1950's he taught himself how to program.

When he married in 1957, the Dutch marriage rites required that he state his profession and he attempted to say he was a programmer. The municipal authorities in Amsterdam did not accept his answer on the ground that there was no such profession.
Throughout Dr. Dijkstra's career, his work was characterized by elegance and economy.

Copyright 2002 The New York Times Company ""

The shortest path is nice to know if you are paying for the wires to connect customers, or even traces on printed circuit boards, especially if you are concerned with high speed computer circuitry.
In the early 1950's telephone rates were based on the shortest path, and this generated a lot of applied research by Bell Labs mathematicians as to how to find that path. They pointed out to the Federal authorities that it was a very complicated problem to consider all the paths that a phone call might take. Customers such as Cal Tech were shrewd and came up with ways to cut costs. This led to the study of Steiner trees, something we may take on later, if the QL can teach us how to do them. Steiner trees allow for the insertion of intermediate junction points.
The shortest path can also serve as a starting approximation in trying to solve the traveling salesman problem.
To see the MST on your QL take all that we had in GG#29 and then merge in the listing "ALGO422MST_Proc". Un-REMark lines 5835 and 5978 that refer to the Minimum Spanning Tree. Enjoy!

I'm happier now with Steve Poole's Voronoi program. I'll let him have a look at my efforts, and hope to get his comments. Perhaps then an article for QLToday? Some of the trouble had to do with the listing being in a font where I and 1 (eye and one) look alike and even the 8 and B(eight and bee) were similar. The most fun was getting rid of GOTO's and turning the subroutines into PROCedures and FuNctions.

Next time ? Connecting the dots with subsets of "neighbors".

TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets.

Cost..... £24

***************NEW****************

superHermes

A major hardware upgrade for the QL

All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/ keyclick) IBM AT kbd I/f // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software)**£90** (£92/£93)
IBM AT UK layout Keyboard.....**£11** (£13/£15)
Serial mouse .....**£8** (£8.50/£9)
Capslock/scrollock LED .....**£1** (£1.50/£1.50)
Keyboard or mouse lead.....**£3** (£3.50/£3.50)
High speed serial (ser3) lead .....**£4** (£4.50/£4.50)

Hermes available for £25 (£26/£27) **Working ser1/2 and independent input, debounced keyboard.**

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.
Cost (incl keyboard lead) .....**£53** (£54/£55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thorn-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating system/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+ Hermes) & built in Multibasic.

First upgrade free. Otherwise send £3 (+£5 for manual if requd). Send disk plus SAE or two IRCs

MKI...£40 (£41/£43) **MKII...£65** (£66/£67)

MINERVA RTC (MKII) + battery for 256 bytes ram. CRASHPROOF clock & I²C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL

A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage - it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second.

Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off **RomDisq** at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq.....**£39** (£40/£41)
4mbytes RomDisq.....**£65**(£66/£67)
8 mbytes RomDisq.....**£98** (£99/£100)
Aurora adaptor.....**£3** (£3.50/£4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/ Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost ..... £34 (£35/£36)

I2C INTERFACES

Connects to Minerva MKII and any Philips I²C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)
2 amp (for 8 relays, small motors) ..... £40 (£43/£44)
4 amp total (for motors etc) ..... **£45** (£48/£50)
Relays (8 3a 12v 2-way mains relays (needs 2a power driver).....**£25** (£28/£29)
Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required.....**£25** (£27/£28)
Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue outputs (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting.....**£30** (£31/£32)
Temp probe (-40°C to +125°C).....**£10** (£10.50/£11)
Connector for four temp probes.....**£10** (£10.50/£11)
Data sheets.....**£2** (£2.50/£3)
Control software & manual (for all I/F).....**£2** (£2.50/£3)

QL SPARES

Keyboard membrane .....**no longer on sale**
1377 PAL .....**£3** (£3.50/£4)
Circuit diagrams .....**£3** (£3.50/£4)
68008 cpu or 8049 IPC.....**£8** (£8.50/£9)
8301/8302 or JM ROM or serial lead.....**£10** (£10.50/£11)
Power supply (sea mail overseas).....**£12** (£17/£21)

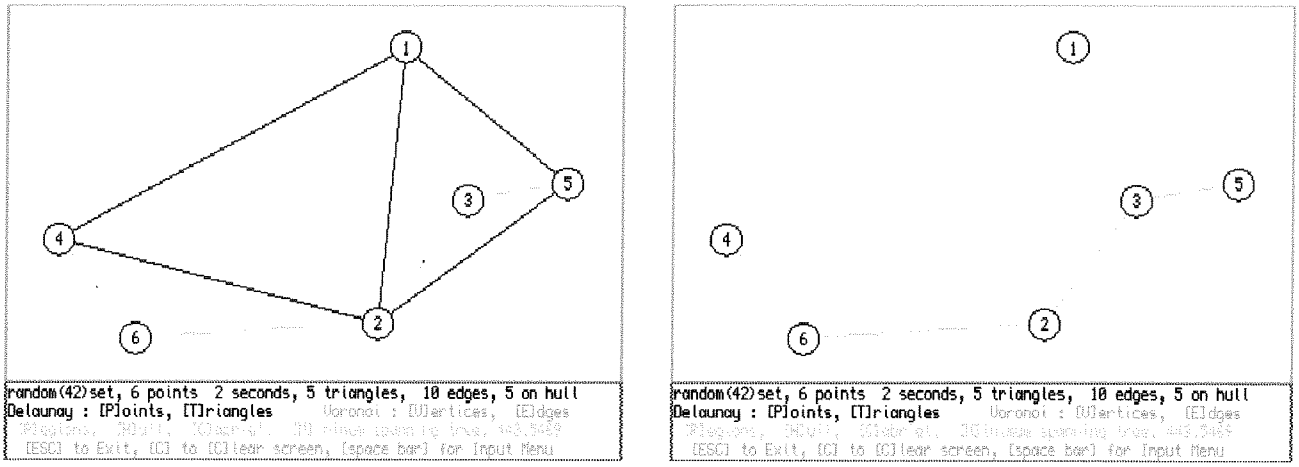
Prices include postage and packing (Airmail where applicable). Prices are: UK (Europe /Rest of world). Payment by cheque drawn on bank with UK address,/ postal order or CASH! I can no longer accept card payments as UK only does PDQ transaction. SAE or IRC for full list and details

10 May 01

29 Longfield Road, TRING, Herts, HP23 4DG
Tel: 01442-828254 Fax/BBS: 01442-828255
tony@firshman.demon.co.uk http://www.firshman.demon.co.uk

```
107 REMark ALGO422MST_Proc for GG#30 Aug. 17, 2002
11260 :
11270 REMark ALGO422MST_Proc
11280 REMark HL Schaaf August 17, 2002
11290 REMark for GG#30
11300 DEFine PROCedure MinSpanTree
11310  LOCal i,j,k
11320 REMark Algorithm 422 CACM 'Minimum Spanning Tree'
11330 REMark adapted for QL S*BASIC
11340 REMark initialize the edges
11350  DIM DM(N,N)
11360  FOR i = 1 TO N-1
11370   FOR j = i+1 TO N
11380    DM(i,j) = SQRT((P(i,1)-P(j,1))^2+(P(i,2)-P(j,2))^2)
11390    DM(j,i) = DM(i,j)
11400   END FOR j
11410  END FOR i
11420 :
11430 REMark Initialize node label arrays
11440  DIM MST(2,N-1)
11450  CST = 0
11460  DIM NIT(N)
11470  DIM JI(N)
11480  DIM UI(N)
11490  NITP = N - 1
11500  KP = N
11510  IMST = 0
11520 :
11530  FOR i = 1 TO NITP
11540   NIT(i)=i
11550   UI(i)=DM(i,KP)
11560   JI(i) = KP
11570  END FOR i
11580 :
11590 REMark update labels of nodes not yet in tree
11600  REPeat update
11610   FOR i = 1 TO NITP
11620    NI = NIT(i)
11630    D = DM(NI,KP)
11640    IF (UI(i) > D) THEN
11650     UI(i) = D
11660     JI(i) = KP
11670    END IF
11680   END FOR i
11690 :
11700 REMark find node outside tree nearest to tree
11710   UK = UI(1)
11720   FOR i = 1 TO NITP
11730    IF (UI(i) <= UK) THEN
11740     UK = UI(i)
11750     k = i
11760    END IF
11770   END FOR i
11780 :
11790 REMark put nodes of appropriate edge into array MST
11800   IMST = IMST + 1
11810   MST(1,IMST)=NIT(k)
11820   MST(2,IMST)=JI(k)
11830   CST = CST + UK
11840   KP = NIT(k)
11850 :
```

```
11860 REMark delete new tree node from array
11870   UI(k)=UI(NITP)
11880   NIT(k)=NIT(NITP)
11890   JI(k)=JI(NITP)
11900   NITP = NITP-1
11910   IF NOT(NITP) : EXIT update
11920  END REPeat update
11930 :
11940 REMark now draw it all
11950  INK 2
11960  FOR i = 1 TO N-1
11970   LINE P(MST(1,i),1),P(MST(1,i),2) TO P(MST(2,i),1),P(MST(2,i),2)
11980  END FOR i
11990  cst$=CST
12000 END DEFine MinSpanTree
12010 :
12020 DEFine FuNction MAX(a,b)
12030  RETurn a*(a>b)+b*(b>a)+a*(a=b)
12040 END DEFine
12050 :
12060 DEFine FuNction MIN(a,b)
12070  RETurn a*(a<b)+b*(b<a)+a*(a=b)
12080 END DEFine
12090 :
12100 REMark end of listing ALGO422MST_Proc
```



Screens are Delaunay with MST and numbered points, and MST and numbered Points.

Just a little Star-Gazing?

Geoff Wicks

QL-Today is now over six years old. During its life it has matured to become more professional in both content and appearance, but it still has one serious deficiency. Every self-respecting English language publication has an astrology column giving predictions for the readers. "A black influence in your life consumes much of your time."

I am, of course, being facetious, but I wonder how many readers know there is a good astrology program for the QL. So good that most of us would not understand it. The program does not produce the newspaper type predictions we are familiar with, but calculates the positions of the planets that astrologers need to make their predictions. The program is justly called "Professional Astrologer".

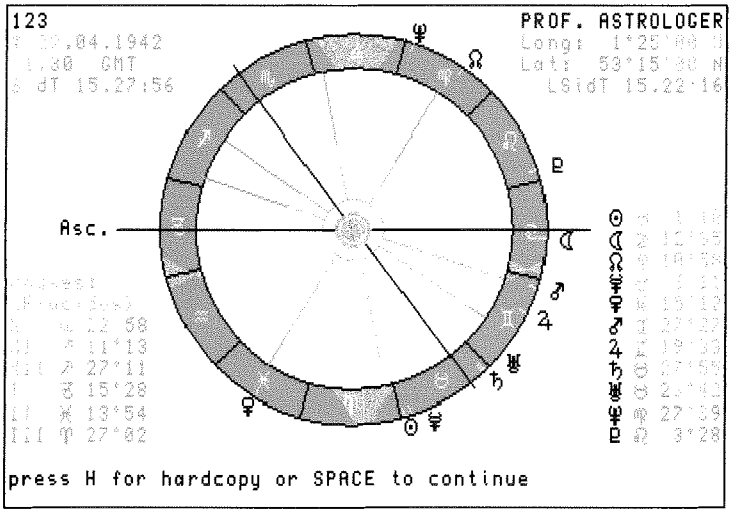
"Professional Astrologer" was often sold with its sister program "Professional Astronomer". The latter name was a joke. All the program did was show the positions of the sun, moon and the planets. OK, it was clever, because it could do this for any time or date for any place on earth and all within the restrictions of a microdrive cartridge and a 128K black box, but at the time there were public domain PC programs that

did much more. They displayed not only the planets, but also the stars. Any astronomer would have no doubt that the PC had the better product.

Earlier this year, however, "Professional Astronomer" gained an advantage on its PC counterparts when there was an interesting and unusual conjunction of several planets. Professional Astronomer enabled me to display then on the screen, and then look out of the window to find them. This would have been less easy with a PC program because I would have been overloaded with information. The numerous stars displayed on my screen would have made it difficult to identify the planets. The simplicity of Professional Astronomer gave it an advantage over more complicated programs.

If we look at the major QL developments over the last couple of years they have been in hardware, not software, with the Q60 as the most important. There has been a continuous and parallel development of QPC2, but, strictly speaking, this is not QL software, but "pseudo-hardware" in that provides a platform on which we can run our QL programs. SMSQ/E has also been improved, most noticeably with the new colour drivers, but again this is not strictly QL software, but an operating system on which we run QL programs. New QL software applications have remained conspicuously absent. What is the point of improving our hardware if we have no programs to run on it?

If we ask the average QL user what new software he would like, he will often be lost for words. Eventually he will come up with suggestions such as a vector word processor, email access, Internet browsers or a major graphics program. There is just one snag. Most of these things are not going to come. At least not unless we can find a benefactor who will pay a professional programmer to write them. There are good professional programmers in the QL community and quite a few competent amateur ones, but they all have to earn a living and this limits the time they have for QL programming. This may appear a sombre picture, but the situation could be different if we changed our way of thinking. When we talk about software we too often think in terms of PC software. If we fight the PC on PC terms we shall lose the battle. If we fight it on QL terms we could just win. What then should be our QL terms?



I began this article by writing about "Professional Astrologer" for a good reason. Even in the QL heyday this program was probably too specialist to have sold many copies. However its history is interesting.

The program was written by Elmar J. Duensser, a computer scientist, who does not believe in astrology, but whose father was a professional astrologer. He was assisted by his wife Hiroko Nakakura-Duensser who entered 15,000 items of astronomical data. She does not believe in astrology. The final work on the program, to make it suitable for commercial release, was done by Freddy Vachha of Digital Precision. His father was an astrologer, but he does not believe in astrology. Finally the program consultant was an astrologer, Janet Augustin, who was the only person in the team who believes in astrology.

In short a team of highly skilled people with an intimate knowledge of their subject produced a QL program of the highest quality, but which would be only of minority interest. It is a fascinating program worth looking at even if you do not believe in astrology. The chances are that you will not understand most of what it does.

There are astrology programs for the PC, but probably nothing like Professional Astrologer. They may have impressive graphics, sound and similar features, but they assume the user is more interested in the result than the theory. The "science" behind astrology can be left to the experts. All the user wants is a prediction of his future.

Now take spellchecking. PC word processors may well have impressive spellcheckers that have more possibilities than the QL ones, but they make similar assumptions about the user. The mechanics of spellchecking is best left to the experts and the user is only interested in the results. This, surprisingly, can give the QL an advantage.

New Millennium.
New Century.
New QL.

What more could you want?

Features

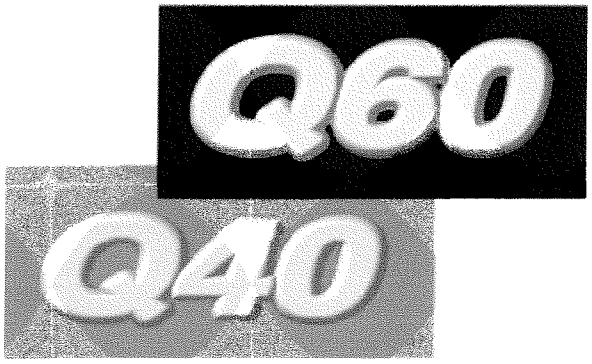
- ✗ Q40i : 68040 CPU, 40 MHz, MMU, FPU
- ✗ Q60/60 & Q60/66: 68060 CPU, 60/66 MHz, MMU, FPU
- ✗ Q60/80: 68LC060 CPU, 80 MHz, MMU
- ✗ 68060 superscalar architecture, dual execution units
- ✗ Up to 160 BogoMIPS performance for QDOS+SMSQ/E
- ✗ 16 to 128 MB RAM, PS/2 module sockets
- ✗ 256 kB ROM (mainboard supports up to 1024 kB)
- ✗ Highspeed 32 bit graphics, plus original QL modes
- ✗ Up to 65536 colours at 1024 x 512 pixel resolution
- ✗ Multisync monitor output (15 pin HD connector)
- ✗ PC Keyboard interface (DIN)
- ✗ 20 kHz Stereo sound
- ✗ Battery buffered clock, 2 KB nonvolatile RAM
- ✗ Controller for 2 floppies and 2 IDE harddisks or CDROM
- ✗ 2 Serial ports with 115200 Baud, Parallel port (on I/O card supplied with mainboard)
- ✗ Hardware extension slot supports ISA cards
- ✗ Fits directly into AT Minitower or other standard case
- ✗ +5V / +12V power supply
- ✗ No tinkering, no parts from original QL needed
- ✗ Mainboard size 8.2 x 6.3 inch
- ✗ Can boot SMSQ/E in a few seconds, directly from ROM
- ✗ Three different operating systems available! SMSQ/E, QDOS Classic, Q60 Linux

Prices

Mainboard	
Q40i	**£ 284.00**
Q60/60	**£ 390.00**
Q60/66	**£ 463.00**
Q60/80	**£ 614.00**
RAM	
16 MB**	**£ 17.00**
32 MB	**£ 34.00**
64 MB	**£ 36.00**
80 MB	**£ 53.00**
128 MB	**£ 72.00**
I/O Card (FLP,IDE,SER,PAR)**	**£ 14.00**
Operating System	
SMSQ/E for Q40i/Q60**	**£ 10.00**
QDOS Classic for Q40i/Q60*	**free**
Q60 Linux CD	**£ 15.00**
Ethernet Card 10 Mbit/s	**£ 17.00**
Cased assembled computers	
Type A (Mainboard in case)	**£ 40.00**
Type B (Fully built system with mainboard, 1.44 MB Floppy, 20 GB Harddisk, 54xCDROM)	**£ 152.00**
Preinstalled software package	
QPAC1, QPAC2, FiFi, QD, PROWESS and much more, over £100 worth	**£59.00**

** Standard parts, always required for operational mainboard *on mainboard support disks

Shipping and handling is extra. Prices may change due to semiconductor costs or exchange rates. All mainboards with VGA lead, sound adaptor, support disks and manuals. Please note: The Q60/80 is not available with floatingpoint coprocessor.



Website and technical information:

www.q40.de

Email: info@q40.de

D&D Systems

P.O. Box 5813, Ripley, Derbyshire, England DE5 9ZR
Tel. +44 (0)1773-740170, FAX +44 (0)1773-748399
Email: sales@q40.de

Financially assisted by a loan from QUANTA

Take the power back in your hands.

You might think it is the simplest thing in the world to get hold of a simple list of English, French, German or Spanish words. In practice it is quite difficult. Every few months I get an email from a university student or researcher requesting such a list. Usually they have never heard of the QL and have spent hours searching the web for a simple text list of words until they came across the Just Words! web page.

In a similar vein I have a vague memory of someone telling the story at Manchester of a QL-er who for many years used the Psion suite to score and collate the results of competitions at his local gardening show. It proved almost impossible to adapt his methods for Excel.

The QL has software possibilities that the PC does not, but we must think as QL-ers and not PC-ers. We can produce simpler programs, smaller production runs and cater for minority interests in a way that is almost impossible on the PC. These are the QL software strengths that we must build on.

Last year I released my program, QL-Rhymes. It was of such specialist interest that I had no idea how many copies it would sell. When asked by other traders, I replied I would be happy if I sold 10 and ecstatic if I sold 20. A year later I am very near to being happy although ecstacy will take longer. I am expecting a similar situation with my new program "Auto-Graph". But both help to keep Just Words! alive, and I suspect the future of Just Words! may lie in similar small scale projects.

I invested hours in these programs, but my labour costs are not reflected in the price. Both programs gave me a stimulating challenge, since in each case I found my original ideas would not work at all or not as efficiently as I would have liked. This was a learning experience I used to develop techniques that would work. If nothing else, I had a sense of satisfaction and achievement when I had finished.

These were my challenges over the last couple of years, but what were yours? Have you achieved anything that could be of use to other QL-ers? You may not be the most expert of programmers, but you may be using your QL in a way that would interest other users. It may be that your idea could be developed by a software writer to make it more generally available.

In the QL world we do our own thing, both as a group and as individuals. Why then must we try to slavishly follow the PC when we think about software and not go our own way?

Which Machine Article - Reply

Al Boehm

As promised in the previous issue, the reply from Al.

In QL Today May/June 2002 p54, Simon Goodwin points out a limitation in finding which machine is being used. A program in the Rich Mellor's et al. S(uper)Basic Manual appears to be able to do this with recognition for various Atari and Thors, the QXL, GC, SGC, QPC, etc.

On the Q-emuLator it says:

```
STANDARD QL - ROM VERSION JSU
ROM VERSION - JSU
OS VERSION - 1U10
PROCESSOR - 68008
```

and similar for QLAYW. Thus, it can not distinguish between

```
1 REMark This example needs the FN Toolkit and Toolkit II to
  be loaded
100 v$=VER$:q$=QDOS$
110 processor$='00':FPU=0:em_type=0
120 IF v$<>'HBA'
130   m_type=PEEK(SYS_BASE+HEX('A7'))
140   em_type=m_type && BIN('11100000')
150   m_type=m_type && BIN('00011111')
160   SELect ON em_type
170     =1:em_type=3
180     =2:em_type=1
190     =4:em_type=2
200   END SELect
210   IF v$='JSL1'
220     PRINT 'MINERVA FITTED'
230     m_type=100 : processor$='08'
240   ELSE
250     IF m_type=0
260       IF PEEK (SYS_BASE+HEX('84'))<>0
270         IF q$(1)='4':PRINT 'THOR 1 Computer';
280         IF q$(1)='5':PRINT 'THOR 20 Computer';
290         IF q$(1)='6':PRINT 'THOR XVI Computer';
300         IF q$(1) INSTR '456'=0:PRINT 'UNKNOWN THOR Computer'
310         PRINT ' v';q$:STOP
320       END IF
330     END IF
340   END IF
350 ELSE
360   m_type=MACHINE:processor$=PROCESSOR
370   FPU=processor$(2):processor$=processor$(1)&'0'
380   em_type=DISP_TYPE
390 END IF
400 extra_chip=m_type MOD 2
410 m_type=(m_type DIV 2)*2
```

```
420 SELect ON m_type
430   =0: IF v$='HBA'
440     PRINT 'ATARI ST / STM / STF / STFM';
450     ELSE :PRINT 'STANDARD QL - ROM VERSION ';v$ :
        processor$='08'
460     END IF
470   =2: PRINT 'MEGA ST or ST / STM / STF / STFM with REAL-TIME
        CLOCK';
480   =4: PRINT 'ATARI STACY';
490   =6: PRINT 'ATARI STE';
500   =8: PRINT 'MEGA STE';
510   =10: PRINT 'GOLD CARD';
520   =12: PRINT 'SUPER GOLD CARD';
530   =16: PRINT 'FALCON';
540   =24: PRINT 'ATARI TT 030';
550   =28: PRINT 'QXL'
560   =30: PRINT 'QPC'
570 END SELect
580 SELect ON extra_chip
590 =0: PRINT
600 =1:
610   SELect ON m_type
620   =0,2,4,6,8,16,24:PRINT ' with BLITTER'
630     SELect ON em_type
640       =0: PRINT 'Original QL Emulator FITTED'
650       =1: PRINT 'Extended Mode 4 Emulator FITTED'
660       =2: PRINT 'QVME Emulator FITTED'
670       =3: PRINT 'Monochrome Display Only'
680     END SELect
690   =REMAINDER :PRINT ' with HERMES'
700   END SELect
710 END SELect
720 PRINT 'ROM VERSION - ';v$
730 PRINT 'OS VERSION - ';q$
735 IF m_type<>30
740   PRINT 'PROCESSOR - 680';processor$;
745 ELSE
750   PRINT 'INTEL PROCESSOR';
755 END IF
760 SELect ON FPU
765   =0 : PRINT
770   =1 : PRINT 'with Internal MMU'
780   =2 : PRINT 'with 68851 MMU'
790   =4 : PRINT 'with Internal FPU'
800   =8 : PRINT 'with 68881 or 68882 FPU'
810 END SELect
```

these two emulators. On the other hand, from a programmers viewpoint, these emulators have very minor differences when configured with the same rom. But there are differences. For example, QLAYW v.90c has win1_ but can not handle flp1_. On the Q-emuLator, flp1_ can be used but points to the same directory as win1_.

Since it might be important in some application to know, I hope Daniele and Jan consider some type of identifier accessible from SuperBasic in their emulators.

In the Manual's program, the toolkit FN apparently is needed for Sys_base (same as Sysbase DIY vol Q) and for QDOS$ which is also found in Tinytoolkit and BTool. QDOS$ is identical to VER$(1) found on Minerva and SMS v2.50+

The listing printed together with this article comes, as said before, from Rich Mellor's SBASIC/SuperBASIC Reference Manual - thanks, Rich, for the permission to print it.

Which Machine Article - Reply

Daniele Terdina

To detect Q-emuLator, you can call TRAP #1 with D0.L=-26 and D1.L=0. If the trap returns with no error (D0.L=0) then you are running on Q-emuLator, otherwise it's something else.
This works for the Windows version, but I'm not sure about Q-emuLator for Mac.

There is no SuperBASIC keyword to do the same, but it would easy for someone interested in detecting Q-emuLator to write a tiny extension to detect it through a call to this TRAP.

And another Reply

Bob Spelten

In your QLToday articles on testing for hardware I missed one point that I test for in my boot program and that is who is doing the pointing? Serial Mouse, Hermes, SuperHermes, Qimi or PS/2.
I use the IPCVER$() function from the Hermes _ext file, to test for Hermes or SuperHermes. Jochen Merz once told me to PEEK_W(114588) to test for Qimi. Reading this address does not always give the same result but when Qimi is present the value is always negative else it is positive. SuperHermes is enabled on my Aurora's so I don't know if it works with Qimi enabled but it finds the right mouse-IF on my QL-SGC with Qimi. Maybe there is a better way?

My BOOT tests for 5 different mice roaming my desk: No or PS/2, Qimi, Hermes & Serial, Hermes & Qimi, SuperHermes. I then know which drivers to LRESPR or not to LRESPR. I have one BOOT for QL-Trump, QL-SGC, Aur-SGC, QXL2 or QPC2 and do most testing before I load SMSQ/E. I save the result to a file and read this back after SMSQ/E has started to guide the rest of the BOOT process.

QLTdis - part 7

Norman Dunbar

As ever, we start this exciting instalment with a few corrections to the previous article. In QLToday Volume 6, issue 5 there are some comments in the code for the 'decode a register list' routine, about half way down the page referring to Andy Pennell and asking 'Why is this required'. Delete those comments entirely - they are no longer required.

This code just below those comments:

```
rl_531  bsr     slash           ; Add a spare slash
        bne.s   rl_560          ; Then remove it again
```

Should be changed to the following:

```
rl_531  bsr     slash           ; Add a spare slash
        bne.s   rl_560          ; Then test for end
```

So far, so good - I've only had to change comments! Just below the above code, we currently have the following:

```
rl_532  subq.b  #1,d1           ; Down to the next bit
        btst    d1,d0           ; Test it
        bne.s   rl_531          ; Next bit is clear -
                                ; add a slash
```

Which has a slight bug in the code, change it to the following:

```
rl_532  subq.b  #1,d1           ; Down to the next bit
        btst    d1,d0           ; Test it
        beq.s   rl_531          ; Next bit is clear -
                                ; add a slash
```

Obviously, if we test a bit and it is not set, then the Zero flag will be set, so the condition I should be using is the 'beq' one and not 'bne'. It makes a slight difference believe me!

And, hopefully, finally, please check that the following two lines are correct in your code. I have made a change at some point to these lines but cannot find reference to them in any of the past articles - I'm just making sure here:

```
type_14 dc.w    $f130,$c100,14,t_exg-op_table

type_15 dc.w    $f038,$0008,15,t_movep-op_table.
```

In volume 7 issue 1, there is one small problem in the sub-routine to extract an index register from an effective address. The code looks like this:

```
index_reg   btst    #15,d0      ; Address or data
            beq.s   ir_data     ; Clear = data
            bsr     aaaa        ; Do an address register
            beq.s   ir_reg      ; Done
```

It should be changed to the following:

```
index_reg   btst    #15,d0      ; Address or data
            beq.s   ir_data     ; Clear = data
            bsr     aaaa        ; Do an address register
            bra.s   ir_reg      ; Done
```

The last line is obviously wrong as we want to skip over the code to add a data register to the output buffer if we have just processed an address register.

Onwards ....

Ok, those are all the code problems that I was able to find, so on we go with the long awaited routines to disassemble instruction types 17 onwards - we are getting close to the end, and so we should, we've been doing this since 1998! In fact, it was Volume 5 issue 6 when we did the type 0 to type 16 instructions.

So lets get on and do the next few instruction types, starting with the type 17, which are a pretty mixed bunch of instructions ranging from 'MOVE SR,‹EA›' to JSR ‹EA›'.
These should be typed into DISS_ASM following on from the type 16 decoding routine that we did all those months ago.

We start off by checking for MOVE SR,‹EA› and MOVE CCR,‹EA› if we have got either of these then we are ok in the buffer so far. Everything else gets a space added. Then we restore the op-code from D7.W and decode the effective address for all instructions. Again we test for MOVE‹ea›,CCR or MOVE‹EA›,SR and if we catch one of these, we add the appropriate text to the buffer and exit the routine.

There are quite a lot of different instructions in this family, but pick one and try to follow it through until you understand what I'm doing.

```
*-------------------------------------------------------------------------
* TYPE 17 instructions
*-------------------------------------------------------------------------
dtype_17    andi.w  #$ffc0,d0       ; Do some extra masking
            cmpi.w  #$40c0,d0       ; If MOVE SR,‹ea› - do nothing
            beq.s   t17_all         ; Skip
            cmpi.w  #$42c0,d0       ; If MOVE CCR,‹ea› - do nothing (yet)
            beq.s   t17_all         ; Skip
            bsr     space           ; Add a space next

t17_all     move.w  d7,d0           ; Restore the op-code again
            moveq   #2,d5           ; All are word sized instructions
            bsr     eff_addr        ; Decode the effective address part
            andi.w  #$ffc0,d7       ; More masking - on D7 this time !
            cmpi.w  #$44c0,d7       ; CCR instructions ?
            bne.s   t17_sr          ; No, skip
            bsr     comma_ccr       ; Add ',CCR'
            bra     p_hex           ; Finished with CCR stuff

t17_sr      cmpi.w  #$46c0,d7       ; SR instructions ?
            bne     p_hex           ; NO - finished with these
            bsr     comma_sr        ; Add ',SR'
            bra     p_hex           ; And finished with SR stuff
```

Next is type 18, which includes the 'ADDI', 'EORI','CLR' and others instructions. Check back in previous articles for the full list. It is quite simple to decode anyway, the buffer already has most of the instruction, all we do here is decode the size and the effective address and then exit via the hex printing routine as usual.

```
*-------------------------------------------------------------------------
* TYPE 18 instructions
*-------------------------------------------------------------------------
dtype_18    bsr     size_decode     ; Decode the size, set D5 and add a space
            btst    #14,d0          ; Check for immediate data or not
            bne.s   t18_set         ; Skip the data bit if set
            bsr     sub_mode4       ; Process immediate data
            bsr     comma

t18_set     bsr     eff_addr        ; Do effective address
            bra     p_hex           ; Done
```

Type 19 are a small group, consisting of the 'ADDQ' and 'SUBQ' instructions only. All we do here is decode the size - .B, .W or .L - followed by the immediate data and finally the effective address itself.

```
*-------------------------------------------------------------------------
* Type 19 - ADDQ and SUBQ
*-------------------------------------------------------------------------
dtype_19    bsr     size_decode     ; Decode the size and add a space
            bsr     hash_dollar     ; Then '#$'
            andi.w  #$0e00,d0       ; Keep bits 9 to 11 only
            bne.s   t19_data        ; Data is not zero - skip
            bsr     eight           ; Add it to the buffer
            bra.s   t19_comma       ; Skip

t19_data    lsr.w   #8,d0           ; Bit 9 becomes bit 0
            lsr.w   #1,d0           ; Can only do 8 bits at once !
            move.b  d0,d4           ; Copy to d4 to add to the buffer
            bsr     d4_hex_b        ; Add to the buffer

t19_comma   bsr     comma
            move.w  d7,d0           ; Reload the op-code
            bsr     eff_addr        ; Extract the effective address
            bra     p_hex           ; Done
```

The type 20s are the bit manipulation instructione where the bit number is specified in immediate data. BCLR #1,D0 for example. This again is quite simple to decode especially as we have a sub-routine to do the hard work of extracting the instruction for this and the type 21 instructions.

All we have in the buffer at this point is a 'B', so we branch off to the sub-routine to get the other three characters of the instruction (CLR, CHG, SET or TST) followed by the requisize space, then extract the data part, a comma and finally the effective address. It all seems to be getting much simpler now - what can possibly go wrong ?

```
*-------------------------------------------------------------------------------
* TYPE 20 - bit manipulation instructions with the bit number held in data.
*-------------------------------------------------------------------------------
dtype_20    bsr     bit_op          ; Finish the instruction & trash D0
            bsr     hash_dollar     ; Some data coming up
            move.w  (a6)+,d4        ; Fetch the data
            bsr     d4_hex_b        ; Extract the lowest byte
            bsr     comma
            move.w  d7,d0           ; Reload the op-code
            moveq   #1,d5           ; Preset an op-code size (pretty meaningless)
            bsr     eff_addr        ; Extract the effective address
            bra     p_hex           ; Done
```

Much of the work for this set of instructions has already been done. There is the same sub-routine as above to finish the instruction in the buffer, and then all we have to do is add a data register and the effective address.

```
*-------------------------------------------------------------------------------
* Type 21 - BTST BCLR with the bit number in a data register
*-------------------------------------------------------------------------------
dtype_21    bsr     bit_op          ; Finish the instruction & trash D0
            bsr     dddd            ; Needs a data register
            move.w  d7,d0           ; Reload D0 with the op-code
            bsr     dest_reg        ; Extract the destination register
            bsr     comma
            moveq   #1,d5           ; Preset an op-code size (pretty meaningless)
            bsr     eff_addr        ; Extract the effective address
            bra     p_hex           ; Done
```

If you remember the type 22s you will know that they consist of the MUL, DIV (signed and unsigned) and CHK instructions and that they all look something like `'MULU‹EA›,Dn'` and this next bit of code is so simple in that the comments basically say it all.

```
*-------------------------------------------------------------------------------
* TYPE 22 instructions
*-------------------------------------------------------------------------------
dtype_22    moveq   #2,d5           ; Preset WORD sized operation
            bsr     eff_addr        ; Effective address decode
            bsr     comma_d         ; ',D' next
            bsr     dest_reg        ; Do the destination register
            bra     p_hex           ; Done
```

Type 23 consists only of the LEA instructions and is decoded quite simply using the next 5 lines of code. Still seems quite simple doesn't it. I'm waiting for the 'gotcha' - there has to be one.

```
*-------------------------------------------------------------------------------
* TYPE_23 the LEA instruction
*-------------------------------------------------------------------------------
dtype_23    moveq   #4,d5           ; Preset LONG sized operation
            bsr     eff_addr        ; Effective address decode
            bsr     comma_a         ; ',A' next
            bsr     dest_reg        ; Do the destination register
            bra     p_hex           ; Done
```

Type 24 includes the ADDs, SUBs, ADDAs and SUBA instructions. There is more to decoding these instructions but again, it is fairly simple. All the following code does is check to see if one of the Address addition or subtraction instructions is found by checking bits 6 & 7 of the op-code word. If they are both set, then this is an 'xxxA' instruction and has the form `'SUBA ‹EA›,An'`.

Note that the routine size_d0 actually returns the size in bytes (1, 2 or 4) because it multiplies the two

DILWYN JONES

QL P.D. SOFTWARE LIBRARY SERVICE

The QL PD Library consists of:

- Over 80 disks assorted PD programs, 15 disks of over 100 games, and 46 clipart disks
- QL manuals and information on disk – replacement manuals for many QL hardware products
- QL Hackers Journal & ClubQL e-zines, 26 demo disks of commercial QL programs
- Back issues of QL Today cover disks, QL emulators for several computers
- Loads of Line Design clipart and fonts, loads of classic literature on disk!
- Lots more than I can list here and the library just keeps growing!
- Just £1 per disk (or 75p if you supply disk) plus postage – see below
- Catalogue available – see below

For a PD software library catalogue, send me a formatted floppy disk, or an email requesting it to be sent to you by email, or download a catalogue from my website (address below). This PD library is run as a non-profit making service for the QL community. All programs in the PD library are PD, Freeware, or Shareware etc, also freely available from other libraries and websites, some also available for free download on my website where space permits.

QL CD-ROMS

QL DOCUMENTATION CD £5.00 – Vast collection of manuals, documents, articles, beginners information etc, ***now version 2, greatly expanded!***
DJ PD LIBRARY CD £10.00 – My entire PD library on just one CD!
QL EMULATORS CD £5.00 – All of the available emulators on one CD-ROM!
QL PD-CDR £5.00 – Gérard Plavec's vast software collection.
LINE DESIGN CLIPART CD £10.00 – Huge collection of Line Design clipart
FAMOUS FACES CD £10.00 – Line Design pictures of the famous!
QL RELIGION CD £10.00 – A collection of Bibles, religious texts, cliparts etc
CD-ROMs supplied in ISO-9660/Joliet format (Rockridge by special request) for use with most systems able to access QXL.WIN media. Q40/Qubide 2 users may be able to use Thierry Godefroy's Atapi/CD drivers (disks GE53 in the library) to access these CDs.

POSTAGE UK/Europe: add £1.00, Rest Of World: add £2.00, per order.
PAYMENT In UK Pounds Sterling currency only please to avoid bank charges for currency conversion. Cheques payable to DILWYN JONES.

Dilwyn Jones, 41 Bro Emrys,Tal-y-bont,Bangor,Gwynedd, LL57 3YT, U.K.
Email:dilwyn.jones@dj.softnet.co.uk
Website: http://www.soft.net.uk/dj/index.html

Note: although run by the editor of QL Today,the PD library is a separate venture with no connection to the publisher of this magazine. Software is supplied free of charge, apart from copying and media charges etc to cover costs and without any warranty other than to the replacement of defective media. In other words, usual PD library terms. E & OE

bit value in bits 6&7 to get the actual number of bytes of data required. If this is an ADDA or SUBA the size will be 3 which doubles up to 6 on exit from size_d0. Guess who spent an hour debugging that little problem then?

If we have found a non 'xxxA; instruction then we decode, using bit 8, which way round the instruction is - either 'XXX ‹ea›,Dn' or 'XXX Dn,‹EA›' and process it accordingly. Again, no major problems.

```
*-------------------------------------------------------------------------------
* TYPE_24 the ADDs and SUBs and ADDAs and SUBAs.
*-------------------------------------------------------------------------------
dtype_24    bsr     size_d0         ; Get the size into D0
            cmpi.b  #6,d0           ; D0 = 6 = ADDA or SUBA
            bne.s   t24_nota        ; Not the A versions
            bsr     aaaa            ; Needs an A in the instruction
            btst    #8,d7           ; Size bit
            beq.s   t24_word        ; Clear = word
            bsr     dot_ell         ; Add the long size to the buffer
            moveq   #4,d5           ; Long sized data
            bra.s   t24_add         ; Skip

t24_word    bsr     dot_w           ; Add the word size to the buffer
            moveq   #2,d5           ; Word sized data

t24_add     bsr     space           ; Then a space
            move.w  d7,d0           ; Reload the op-code
            bsr     eff_addr        ; Extract the effective address
            bsr     comma_a         ; Then a ',A' is added to the buffer
            bsr     dest_reg        ; Here comes the register number
            bra     p_hex           ; All done for ADDA and SUBA

t24_nota    move.w  d0,d5           ; Correct size for effective address decode
            move.w  d7,d0           ; Reload the op-code

t24_type25  bsr     size_decode     ; Add the size specifier to the instuction
            btst    #8,d7           ; Set = ADD Dn,‹ea›
            beq.s   t24_notd        ; Must be ADD ‹ea›,Dn
            bsr     dddd            ; Data register first
            bsr     dest_reg        ; Get the register number
            bsr     comma           ; The a comma

t24_notd    bsr     eff_addr        ; Extract the effective address
            btst    #8,d7           ; Test again
            bne     p_hex           ; Finished with ADD Dn,‹ea›
            bsr     comma_d         ; The instruction needs ',D' next
            bsr     dest_reg        ; Get the register number
            bra     p_hex           ; Finished with ADD ‹ea›,Dn
```

The good thing about the type 25 instructions is that much of the work has already been done above in the type 24s. This means that after a few checks, we simply branch into the type 24 code to finish off the decoding.

Type 25 includes CMPA, EOR and CMP. We test first for the size bits being both set and if so, we must be doing CMPA. Having added it to the buffer, we decode the size in bit 8 and add a '.W' or '.L' accordingly. As with the type 24 instructions ADDA and SUBA, byte sizes are not permitted. Once we have the size appended to the buffer, we exit through the code in type 24 to add spaces, effective addresses and so on.
Even easier is the remaining instructions in this family. All we have to do is place the instruction into the buffer, and let the type 24 code above do the rest of the decoding for us.

I just love re-using code - when it all works out that is !

```
*-------------------------------------------------------------------------------
* TYPE 25 - CMPA, EOR & CMP
*-------------------------------------------------------------------------------
dtype_25    bsr     size_d0         ; Get the size bits into D0
```

```
            cmpi.b  #6,d0           ; D0 = 6 = CMPA.size ‹ea›,An
            bne.s   t25_nota        ; Skip if not doing CMPA instruction
            move.l  #'CMPA',d4      ; We are doing CMPA
            bsr     str_add_1       ; Add it to the buffer
            btst    #8,d7           ; Check for the size
            beq.s   t25_word        ; Skip if word sized
            bsr     dot_ell         ; Add the long size details
            moveq   #4,d5           ; And the size 'size'
            bra.s   t24_add         ; CMPA.L is done, exit via the type 24 code!

t25_word    bsr     dot_w           ; Add the word size details
            moveq   #2,d5           ; And the size 'size'
            bra.s   t24_add         ; CMPA.W is done, exit via the type 24 code!

t25_nota    move.w  d0,d5           ; Correct size for effective address decode
            move.w  d7,d0           ; Reload the op-code
            btst    #8,d7           ; EOR or CMP
            beq.s   t25_cmp         ; Skip if clear (CMP)
            move.l  #'EOR',d4       ; Set is EOR
            bra.s   t25_both        ; Skip

t25_cmp     move.l  #'CMP',d4       ; Doing CMP

t25_both    bsr     str_add_3       ; Add to the buffer
            bra.s   t24_type25      ; Exit via type 24 code
```

So, once again, we are finished for this episode. I shall be continuing from here in the next article - I think it is about time we got this project out of the way and consigned to history. Three and a bit years is a bit long to be working on the same thing in my opinion.

Next time, we'll start with the instructions that have given me the most grief so far, type 26 - the 'Scc' instructions. I have been having problems with these as they turn out to have a similar result to other instructions and are being decoded wrongly.

So far, I have been trying to fix the problem without too many changes to existing code, but it looks like I may have to combine a couple of families of instructions to avoid errors in the output.

See you then!

Drag & Drop with EasyPTR - Part 2

Wolfgang Uhlig

Wolfgang asked us to mention that we split the article in two parts, it was not him. We had to, because first we had no material at all, and when we asked around we got lots of feedback, which gave us a space problem in the previous issue. We often have to do this to make sure the magazine has a good mix - and is not filled by three articles.

```
100 REMark ***** a small EasyPtr-practise
110 :
120 LRESPR win1_projectmenus
130 :
140 OPEN #3,con_
150 MDRAW #3,1
160 INIT
170 REMark ********************************
180 REMark *** start of the main program ***
190 :
200 REPeat main
210 key=MCALL(#3,key,0)
220 SELect ON key
230  =-1:MCLEAR #3:CLOSE #3:STOP
240  =-4:SHOWTIME
250  =2^16 TO 2^32:
260           awnum=key
270           position=MAWNUM(#3,awnum)-1
280           IF awnum=3
290             NEXT main
300           ELSE
310             DRAG_DROP
320           END IF
330  =REMAINDER :NEXT main
340 END SELect
350 END REPeat main
360 REMark *** end of the main program ***
370 REMark ********************************
380 DEFine PROCedure INIT
390 REMark *** define/set variables for
    drag&drop
400 drag%=0:buffer$=""
410 :
420 REMark ** create an array for 20 names
430 DIM names$(19,18)
440 RESTORE
450 FOR i%=0 TO 19
460  READ a$
```

```
470  names$(i%)=a$
480 END FOR i%
490 DATA "Roy","Darren","Tony","Dilwyn"
500 DATA "Jochen","Wolfgang","Marcel","Dietrich"
510 DATA "Al","Dave","Phoebus","Nasta"
520 DATA "Thierry","Jerome","Francois","Claude"
530 DATA "Sjef","Per","Jonathan","Derek"
540 :
550 REMark *** create an array for the
    selected players
560 REMark *** which is empty at the moment:
570 DIM selection$(10,18)
580 :
590 REMark *** an array for 52 weeks
600 DIM weeknumber$(51,2)
610 FOR i%=0 TO 51:weeknumber$(i%)=i%+1:
620 :
630 REMark *** now draw all arrays:
640 MAWDRAW #3,1,names$
650 MAWDRAW #3,2,selection$
660 MAWDRAW #3,3,weeknumber$
670 END DEFine INIT
680 :
690 DEFine PROCedure DRAG_DROP
700 IF drag%
710  IF MTEXT$(#3,key)<>""
720   NEXT main
730  ELSE
740   IF awnum=1
750    names$(position)=buffer$
760   ELSE
770    selection$(position)=buffer$
780   END IF
790   MAWITEM #3,key,,buffer$
800   buffer$=""
810   drag%=NOT(drag%)
820   SPRS #3,0
830  END IF
840 ELSE
850  IF MTEXT$(#3,key)<>""
860   IF awnum=1
870    buffer$=names$(position)
880    names$(position)=""
890   ELSE
900    buffer$=selection$(position)
910    selection$(position)=""
920   END IF
930   MAWITEM #3,key,,""
940   drag%=NOT(drag%)
950   SPRS #3,1
960  ELSE
970   NEXT main
980  END IF
990 END IF
1000 END DEFine DRAG_DROP
1010 :
1020 DEFine PROCedure SHOWTIME
1030 LOCal number
1040 number=MAWNUM(#3\3)
1050 open_over #4,ram1_showtime
1060  print #4,"In week "&number
1070  print #4,"the following players were
     selected:"
1080  print #4,selection$
1090 close #4
1100 END DEFine SHOWTIME
```

In line 120 we load our appendix-file as a resident file. We then open a channel for our menu, (I always take #3 as the first one, don't ask me why), and draw it with **MDRAW**. This command is here in its most simple form. If you had different menus in your appendix-file, there could be MDRAW #3,2 or ...#3,3, you don't have to give the name or even the whole filename of your menu, but only the number in the appendix-file. When "appending" your menu you could choose between fixed or relative which means that the menu appears on the place where it was in EasyMenu_exe when appending (fixed) or where the pointer is at the moment of calling it (relative). You can overrule this by giving MDRAW parameters. MDRAW #3,1,100,50 would for example force your menu to appear at x=100 and y=50. This is only true for the first menu, which determines the outline. Every other menu, drawn later, will be placed within this outline.

We now call the procedure INIT, which you find in lines 380 to 670. Here we first create and set two variables we shall need in the drag & drop routine. Then the three arrays we need, are created and drawn in our application-windows with the command **MAWDRAW**. This command is also used in its most simple form:

```
MAWDRAW #channel, application-window-number,
array-name
```

There are many more parameters that can go with MAWDRAW but they are mainly interesting with three dimensional arrays, which I could deal with next time, if there is interest.

Okay, everything is ready, so let's start our program by opening the main loop in line 200. All actions in a pointer driven application take place within this main loop and are controlled by the most important EasyPtr function **MCALL**. MCALL registers everything that 'happens' within the outline and stores it in the variable you defined for it. In our program this is the variable "key".

MCALL can have different parameters, among other things you can determine the status each menu item will have after clicking on it. For example:
key=MCALL #3,-1,0,-2,0-,3,0,-8,1,9,-1
etc. In our small program, however, the best solution is to always restore the status 0=selectable, after clicking. This is done by the form you find in line 210. In more complex programs where menu items sometimes have to have a different status, this version would not be sufficient. You would then have to learn to use MSTAT or MSTAT%.

CAMBRIDGE Z88

OFFICE/FAX 01494–871319 (EEC) *W.N. Richardson & Co.*

MOBILE: 07808 576118

E-MAIL: WNR@COMPUSERVE.COM

6 Ravensmead
Chalfont-St-Peter
Buckinghamshire SL9 0NB

THE IDEAL PORTABLE COMPANION FOR THE QL, PC AND MAC.

THE CAMBRIDGE Z88 A4 NOTEBOOK

WITH BUILT-IN WORD PROCESSOR, SPREADSHEET, DATABASE, BASIC, CALCULATOR, CLOCK, ALARM, CALENDAR & VT52 TERMINAL.
USES 4XAA ALKALINE CELLS (ca. 20 HOURS) *

WE DO REPAIRS, PART EXCHANGES, AND BUY Z88's & PARTS

NEW! CAMBRIDGE Z88 WITH OZ4+512k Internal Ram £150

ALSO AVAILABLE, 512k RAMPACK £40

DESCRIPTION	UK	USA
CAMBRIDGE Z88 ISSUE4-128K	**£120**	**$180**
CAMBRIDGE Z88 COMPUTER	**£99**	**$130**
☆**RECONDITIONED Z88**	**£60**	**$90**
32k RAMPACK	£18	$25
128k RAMPACK	£38	$50
32k EPROM PACK	£18	$25
128k EPROM PACK	£24	$33
256k EPROM PACK	£55	$75
EPROM ERASER	£32	$43
PARALLEL PRINTER LEAD	£39	$50
SERIAL PRINTER LEAD	£12	$16
* **MAINS ADAPTOR (230vac; 6v,500ma)**	**£10**	
TOPPER (PROTECTIVE COVER)	£12	$16
CARRYING CASE (PLASTIC)	£10	$12

1/2 PRICE STARTER PACK
1 - Z88 ☆ PLUS TRANSFER KIT
+ 32K RAM
+ 32K EPROM
£72 ONLY!!

FILE TRANSFER TO OTHER COMPUTERS		
SPECIAL Z88-QL SERIAL LEAD	£10	$15
COPY DISKS OF QUANTA PROGS IMP/EXP & ARCHIVE EXPORT	£2	$3
PCLINK KIT (For PCs)	£25	$35
Z88 TO MAC KIT	£25	$35
Z88 TO BBC KIT	£25	$35

ALL WITH INSTRUCTION BOOKS, CABLES AND EPROMS AS REQUIRED!

POSTAGE UK UP TO £5. EEC £15. USA £20. OTHER COUNTRIES £30.

ALL THE STOCK IS WARRANTIED FOR 90 DAYS. IN THE EVENT OF REPLACEMENT BEING ARGREED, BUT THE ITEM BEING OUT OF STOCK AT THE TIME, A REFUND WILL BE MADE PROVIDED THE ITEM IS RECEIVED IN GOOD CONDITION.

QL & PC COMPUTER USERS WILL FIND THE CAMBRIDGE Z88 ESPECIALLY USEFUL FOR WORK AWAY FROM THE DESKTOP, WITH TRANSFER PROGS DATA CAN BE SAFELY EXCHANGED WITH THEIR DESKTOP SYSTEM.

W.N.RICHARDSON & CO CONTINUES TO PROVIDE FULL SPARES AND SERVICES FOR SINCLAIR COMPUTERS, QL & THE CAMBRIDGE Z88

From line 220 to 340 we "select" what is going to happen. This is dependant on the value of "key":

All negative values are loose menu items. -1 tells us that the "ESC"-button was clicked and therefore we remove the menu, close the main channel and stop the program. The values for MOVE (-2) and SLEEP (-3) are not necessary because you set these buttons to automatic recognition in Easymenu_exe. Value -4 is our loose item "showtime". Its action is defined in the procedure of the same name and is self-explanatory (except for one thing, see below). The really interesting thing starts in line 250:

In order to manipulate menu items in application-windows (in fact fields in arrays) you have to identify them. They start with 2^16 and end with 2^32. Since these are very unwieldy numbers to deal with you can better use the function **MAWNUM**. It cost me some time to understand this function because the manual was so unclear. However, I think I have understood the most important things and I'll try to explain.

Normally with:
num=MAWNUM(#channel,key)
you get the number "num" of the menu item in the application-window you clicked on. Channel is the main channel and "key" is what you get back from MCALL, in this case really "our key". You also have the possibility to get the coordinates of the menu item, for example:
num=MAWNUM(#channel,key,xpos%,ypos%)
xpos% and ypos% then give us column and line of the chosen menu item. This is, however, more interesting for three dimensional arrays (see above).
Very important is that after using this function, "key" has no longer the value given by MCALL, but has changed to the pure application-window-number!!
The advantage of this is the easy determination in which application-window the action took place. The disadvantage is that you have to define a new variable if you want to keep using "key".

In line 260 I do just that: The new variable "awnum" (application-window-number) is introduced and given the value of "key". This is not yet true in this line, but it becomes true in line 270: "awnum" is now really the awnum, "position" is the field in the array and "key" remains "key", which will be important later.

Attention: the menu items in an application-window always start with number 1. As an array always starts with 0, it has to be:
position=MAWNUM(#channel,key)-1

(There is yet another form of MAWNUM, which I shall comment on in the procedure "SHOWTIME")

In lines 280 to 320 we decide what happens when we click a menu item in one of the three windows. If it is application-window number 3, nothing happens (you can of course change the number by clicking on arrows and/or scrollbar!), and we go to the next main-loop. If it is one of the big windows, the drag & drop routine should start.
In this routine we want to be able to move names from one window to the other or back or within the same window. No name should be overwritten or put in another place twice.

```
700 IF drag%
```
drag% is a variable which, if it is '1', shows that there is still something in the buffer to be placed elsewhere. There are two possibilities: first, there is (already or still) a name in the clicked field. We check this with the function **MTEXT$.**

MTEXT$ is a simple function, with which you can read text out of menu items or info-objects (see EasyPtr-manual). In the case of menu items "num" is the value given back by MCALL, in this case "key". In the case of info objects you have to give info window-number and info object-number and leave out 'num'.

```
710 IF MTEXT$(taste)<>""
```
We don't want to overwrite this text, so...

```
720 NEXT main
```
Or it is empty, then we can drop our buffer there....

```
730 ELSE
```
we first check in which application-window we are and then set the field of the corresponding array to the contents of the buffer.

```
740   IF awfnum=1
750     names$(position)=buffer$
760   ELSE
770     selection$(position)=buffer$
780   END IF
```
At last we bring the contents of the buffer into the menu item using the command MAWITEM, empty the buffer and set drag% to its contrary (0 in this case). We set our sprite back to the original with SPRS and finish our move-action..

```
790    MAWITEM #3,key,,buffer$
800    buffer$=""
810    drag%=NOT(drag%)
820    SPRS #3,0
END IF
```

The syntax of the command ***MAWITEM*** *is identical to* ***MITEM****, except for the fact that it is for menu items in application-windows. Attention: don't forget the second comma after "key" which is necessary for the empty parameter 'underline a letter of the text' when you don't need that. The contents into a menu item set with M(AW)ITEM only become visible after the status of the item is set anew. In our program this is done by the MCALL-function which sets every status to zero after every main-loop.*

SPRS *is a simple command. The description in the manual, however, can lead to some misunderstanding and heavy frustration. It's written there that you can take the name of an appended sprite definition, but this doesn't work well in compiled programs which can drive you mad if you don't know (but you know by now). I recommend to use what is described as "adr" in the manual, but what is not really a memory address, but very simply the number of the sprite in the appendix-file! In our case it's the '1' because we have only one sprite. A '0' sets back to the original pointer. With SPRS #3,1 you set this sprite as pointer for your program, with SPRS #3,0 you restore the original one. Life can be so simple, can't it?*

Now we have to handle the other case, no field had been selected and the buffer is thus empty.

```
840 ELSE
```

In this case we, of course, want something to be in the selected field, empty fields are ignored. (see lines 960, 970)

```
850  IF MTEXT$(#3,key)<>""
```

Look again in which application-window we are. Then put the corresponding contents into the buffer and set the field of the array to 'nothing'.

```
860    IF awnum=1
870     buffer$=names$(position)
880     names$(position)=""
890    ELSE
900     buffer$=selection$(position)
910     selection$(position)=""
920    END IF
```

The empty field has of course to be drawn...

```
930    MAWITEM #3,key,,""
```

drag% becomes its contrary, (thus 1) and the pointer is set to our own sprite

```
940    drag%=NOT(drag%)
950    SPRS #3,1
```

If the field we clicked, was empty, we go to the next main-loop.

```
960  ELSE
970   NEXT main
980  END IF
990 END IF
1000 END DEFine DRAG_DROP
```

That's it already! It wasn't difficult, was it? Of course you could do things differently, this is only a proposal. If you have more than two application-windows to deal with, it's more practical to use a SELect and not an IF for the choice of the right window. Also you could bring the actions of the DRAG_DROP routine right into the main program. It's a question of personal style in the end.

At last we have still one procedure to get the result of the third application-window.

```
1010 :
1020 DEFine PROCedure SHOWTIME
1030 LOCal number
1040 number=MAWNUM(#3\3)
```

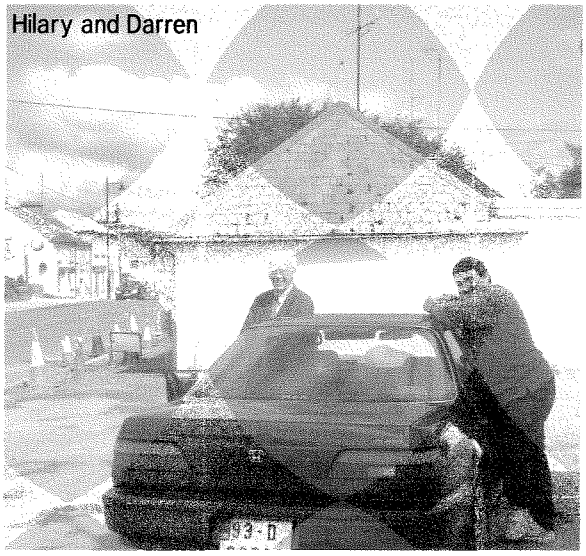
Here you see another variant of the MAWNUM-function. In our program we can neither select a number in application-window number 3, nor edit it. What we can do actually, is change it by clicking the scroll bar or arrows. In order to use the chosen number for an action we have to read it out. This is done by:
number=MAWNUM(#3\3)
In this case you take a backslash instead of a comma, followed by the number of the application-window. Normally you then get the menu item which is on **the left top position of the window!** *In our case, where we have only a two-dimensional array and, above, only one visible field, it's just that one. (We don't need to subtract '1' here, because we did that when creating the array)*

I am not sure whether I could make everything clear as I don't have the English manual of Easy Ptr. Geoff Wicks was so friendly as to check this article so I have good hope. It was a pretty hard piece of work (it's not my mother tongue in the end) and if you are interested in more, please write to me or to QL-Today. If there are no reactions I will take that as a sign of no further interest. If you have questions or comments, please contact me: **wolwol@compuserve.com**

Do you have more listings and explanations for us? Articles about the Pointer-Interface are amongst the most wanted by our readers!

Ireland is CLOSED

Tony Firshman


Hilary and Darren

When Stuart Honeyball (of Miracle Systems fame) heard of the 2nd Irish show organised by Darren Branagh his reply was something like 'must get Old Firshman to cycle again'. In 1997, we cycled with Jeremy Reeves to Laragh.

.... so he and I set off from Yate on our folding bicycles for Bristol Parkway. We started the way we meant to go and had a swift pint in Iron Acton on the way, and Stuart had his first puncture. "I have added Green Slime so it will repair itself", and so it did.

Arrived in good time at the station to find the buffet CLOSED.

I always enjoy going by bicycle on the ferry as one cycles on, and are the first off. I almost ran down a Stena employee who did not hear me coming. We decided to miss the first train and have a MEGA IRISH BREAKFAST. They contain the same everywhere - including at Darren's (is it laid down in statute, Darren?) at a place we remembered from 1997. It was CLOSED. We joined a Honda Gold Wing flock at a place down the road and had the statutory breakfast, including white pudding (a variant of black pudding).

Further down we saw a sign saying:

..... and nothing else.

We then cycled to Wexford for a beer top-up, remembering to 'Yield' at junctions when ordered to. That and the general spaced layout of the houses, shops etc on the road out of Rosslare reminded us very much of America. This was the first trip we had noticed the way the Irish houses are set up - they have a funnel arrangement to the gates. We decided this was to ensure the occupants were guided home after a late night pub session.

Road signs are confusing as some have miles and some Km, but they often do not say which.

We then cycled to Enniscorthy for lunch, and the second of the three trains a day. There we had what must have been the worst pub food I have ever had. A tastless and bright orange hued 'chicken and bacon' Penne arrived, and costing about £5. There was bacon - about 5 tiny pieces, but I could find no obvious chicken. In true British style we duly ate and left without comment. Still the Guinness was good.

We arrived by train at Rathdrum for the final 14 mile cycle to Darren's pad. Unfortunately Stuart's map misses most minor roads, and we went the long and very hilly route. The only road signs were to guest houses. Darren says the Americans nick them! "Is this the road to Clara?" I said. "Yes - straight on" said a helpful local. "Did you mean Lara?" he said. "Yes, where is that then?", I asked. "Straight on".

Stuart had another flat, but the green slime did its job again.

Wicklow Hills are never ending, as I struggle with a low ratio about 50% higher than Stuart's. I think my muscles are about to break.

We arrive in Laragh about 19:30, after some 50 miles of up and over hills. Darren has long gone on his taxiing service, so we tell his parents we will eat in Laragh and do the 3 mile hill climb later.

We arrive at Darren's, negotiate the apparently padlocked gate and set up shop in one of the guest rooms. All is closed for decoration/repairs so there is no toilet or hot water. Still it is very nice to be inside and not cycling.

We set off early for the show to have breakfast there and take the keys. On the way Stuart has two puncture repair stops which means we are very late. Darren collects the keys on the road.

We arrive at the hall at about 10:30 to find a solitary John Hall - the Branagh taxi service is out again picking up punters.

The one unexpected visitor arrives at midday and keeps me busy repairing his QLs. I brought a battery powered 3" TV.

All seven of us went to the local for a liquid lunch. How many pints of Guinness does it take to repair a QL?

Darren had set up a magnificent array of free food for us visitors. After lunch we watched the Qcelt DVD - looks a very professional product. ... but it is, produced by Steve Reyal who does that sort of thing for a living. Pleasantly surprised at seeing myself - but I provided a good clutch of outtakes.
All seven of us (again) had an excellent evening meal in a local restaurant (and more Guinness later). A QLutch of QLers? It was a very pleasant day - pity there were not more people. Well done Darren.

Literally just before we leave, I repair Darren's toilet - I have a pipe wrench in my toolkit (of course). "Do you want to stay a week" I heard Darren's father cry, and he probably meant it.
A pretty uneventful journey home, or so we thought. When we arrived in Fishguard, the connecting train left as we approached the platform - apparently GWT had not told the guard. Coaches were laid on at 3am and we met the same train at Swansea, but over an hour late!

A nice postscript about MY Psion Series 5.
A few years back I entered a free competition at a trade show. It was at the last minute, so the salesman corrected my answers. I persuaded a reluctant Stuart to enter - and he won, so he has always called it 'YOUR Psion' However the screen had packed in and he had failed to repair it. "Get it going and it is yours" so I have and it is now MY Psion - thanks Stuart.

Setting Up QPC2

Roy Wood

This could be sub-titled 'Instructions for the removal of liquid substances from hard bodied embryonic enclosures for female parents whose offspring have already reproduced.' I will, however, offer no apologies to those who think this is so because there are many people who will benefit from setting this out. The manual for QPC2 is pretty good but it only describes the settings and not the reasons why the user may want or need to change them.
Q Branch have sold QPC since it first came out and I have have used it constantly since then. I consider it to one to the best programs I have ever bought and the work that has gone into updating and maintaining it has been consistent and of a high quality.

So where do we start ? Let's begin with a few 'whys'

Whys

When QPC first arrived on the QL it followed in the footsteps of the QXL hardware emulation card. The program and the version of SMSQ/E that it used were both DOS based and would run without any of the Windows routines loaded. In fact, when Windows 95 was released, it was more of a problem if that program was running owing to the strange way that Windows 95 handled dropping into DOS.

The configuration of QPC was handled by a DOS version of the familiar 'Config' program. The QXL version of SMSQ/E had the hard drive locations hard-coded into it and this made the allocation of more than one 'WIN' drive very difficult. QPC would however allow the user to set different locations and names for the 8 allowable 'WIN' devices.
This is, actually, the main reason that I got involved in writing this because many users could not quite understand how to get other 'WIN' drives or, indeed, what these drives were when they were created.

This article will cover that process and touch on a few other useful points. Whenever users upgrade, the programs they are upgrading to often contain new features or changes to the way that the original worked. These changes and innovations can be missed so I hope that even experienced users will gain from this.

The Configuration Window

When you start QPC2 for the first time you will see the Configuration Window. Once you have set the program up to behave how you want it to behave you can hide this away so it will only appear when you call it but it does bear looking at.
The top left window shows the display driver used by QPC2. If you have a twin head graphics card or two graphics cards (and two monitors of course) in the machine you can get QPC2 to appear on the second screen. This would allow you to use both Windows and QPC2 at the same time ! This is something I have not tried out myself but it does have a lot of useful connotations.

Below that there is a rectangular window (Display Modes) showing all of the allowed screen resolutions and beside that are windows which will change the 'Colour depth', 'Resolution' and 'Window Mode'.

You can choose the resolution you want to start QPC2 in by clicking on the appropriate entry in the 'Display modes' box. The values in the 'Resolution' and 'Window Mode' boxes will change to reflect the choice.

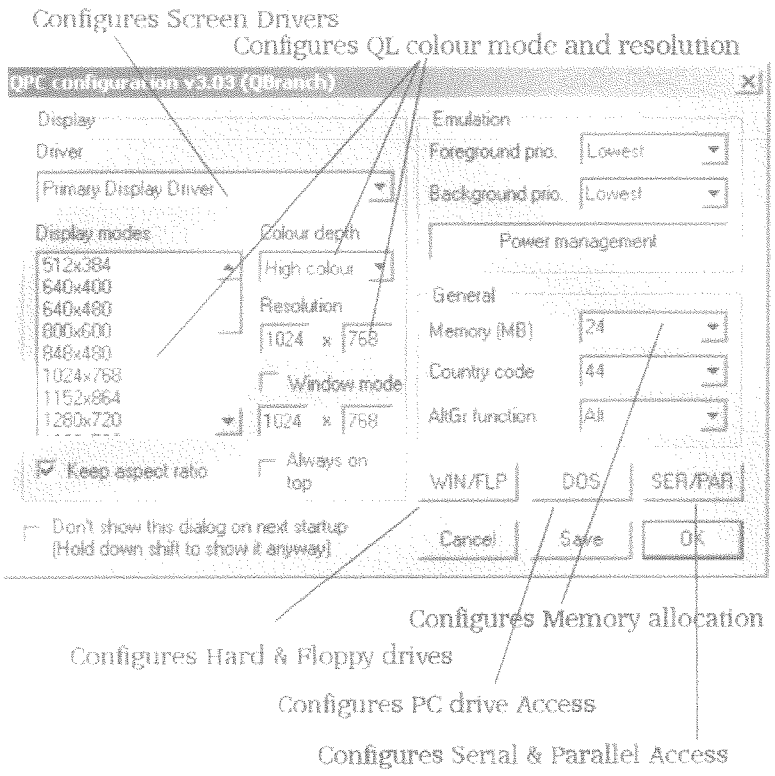
If the Window mode box is unchecked (has no tick in it) then QPC2 will fire up in full screen mode. If you want QPC2 to run in a Window on the screen then click on the square beside the words 'Window mode' and that is how it will start. The choice of which mode to run in does depend, to a degree, on your graphics card and the use you put QPC2 to. You will need a good graphics card to get smooth operation from QPC2 in 'High Colour/Window mode' but it does allow the switching between the emulator and programs on the host system with a single mouse click and you can resize the QPC2 window very easily.

The 'Colour depth' allows you to choose whether QPC2 starts in High Colour or QL Colour Mode. This is done by clicking on the triangle to the right of the window and selecting the mode you wish to use.

On the top right hand side of the Configuration Window there are windows which allow the user to change the Foreground and background priority of the emulation. You can, for instance, raise the background priority so that processes that you leave running on QPC2 while you are switched into Windows get a bit more processor time.

Below this, in the 'General' Section you can set the Memory allocation, Country code and the function of the 'ALT GR' Key.

In earlier versions of QPC2 (I am using v3.03) there was a trade off between the speed of operation and the memory allocated. The higher the amount of memory allocated to QPC2 the slower the file access became. This was because the slave blocks had to be searched before the file was taken from the hard drive. If there was a large block of memory that search was a lengthy one. Innovations introduced with the latest version of QPC2 have slashed the search time and that greatly speeds up the way that programs and DATA files open.



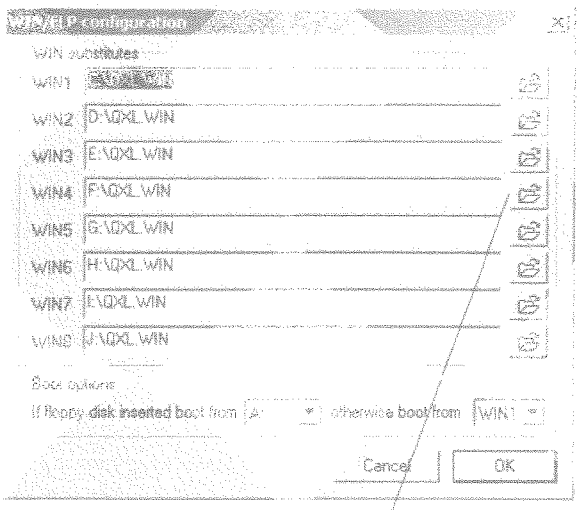
This does mean that you can allow QPC2 a lot of memory but don't forget that it is running on top of another system and that system will slow down and may even freeze if it has not enough memory to work with.

The Country code setting is fairly straightforward and will set the keyboard etc to the country whose country code is shown in the window. These codes are based on the International Dialing codes but only a few are allowed for. Check the manual for more details.

The 'ALT GR function' refers to that funny key beside the space bar. ON most UK PCs this will not have seen too much use and it was provided to access other characters not displayed on the keyboard. Foreign users, especially those in Germany and other countries where they use accented characters, find this key invaluable because it allows them to access characters not available in other other way. You can, therefore toggle this setting to make the key behave how ALT behaves.

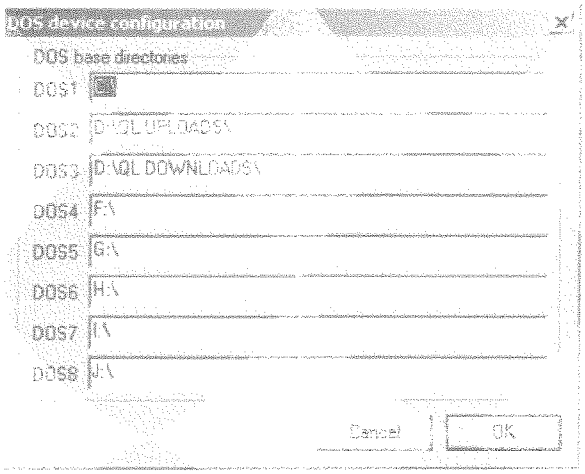
Beneath these settings there is a row of buttons marked 'WIN/FLP', 'DOS' and 'SER/PAR' these are the, all important, drive and port allocation settings and need special attention.

The WIN/FLP Window



The WIN/FLP allocation table

Click on these to find a QXL.WIN file



The DOS drive allocation table

QPC2 can access up to eight WIN drives at the same time. These drives are seen by the emulator as standard hard drives with their own subdirectories but the PC will see them as one large file. This is a very useful thing because it means that, if you have a CD writer or ZIP-drive, you can back up the entire QL WIN drive in a single operation just by copying that file onto another medium. This does need a bit of forethought by the user.

If you have a 100Mb Zip drive then make your QXL.WIN drive no larger than 95Mb. This will mean that you can just copy the QXL.WIN drive from one medium to the other leaving 5Mb free for any files or File Allocation Table that Windows wants to put onto it.

When you get your copy of QPC2 it should come with the drives configured as in the illustration. You can either click on the file name and edit it directly or, if you have a particular QXL.WIN drive you want to use click on the folder icon and open the Windows file manager to look for it. One mistake that people have made is assuming that changing a name or location in this table will create the file they want. All this does is to point QPC2 in a particular location when that drive is called. To create the drives you sill have to use the 'Format' command.(don't worry this will not format your PC's drive - it just creates a file with the name you have chosen at the location you have chosen') For example:

I want to create 'WIN2_' on my ZIP drive so I can copy some data to it and take it to another machine. My PC sees the ZIP drive as 'F'. Here are step by step instructions to do that.

STEP 1: Open the QPC2 Configuration window and edit the window called 'WIN2' to read 'F:\QXL.WIN' (it does not have to be QXL.WIN but it is worth keeping the '.WIN' extension so you know what you are looking for. Remember this is only a name tag for a file. Leaving it as QXL.WIN will make the transfer easier). You do not need to have the ZIP disk in the drive when you do this but it must be there when you come to create the file - logical I know but some people have got this wrong.

STEP 2: Fire up QPC2 and type at the command line:

```
WIN_FORMAT 1
```

When you press ENTER nothing will happen but QPC2 will be set up to create WIN2_. Now type:

```
FORMAT WIN2_95
```

When you hit ENTER you will be asked to enter two letters and entering these will start the format process. A message such as 'Format Failed' will mean that either you have write protection on the drive or you have set something up wrong. Go back to step 1 and check. (This may seem a long winded way to go about it but it does mean that you will never format a drive unintentionally.)

STEP 3: Directory WIN2_ to see if it is there.

STEP 4: Go back to Windows and check the zip disk to see if there is a file on 'F' called 'QXL.WIN'

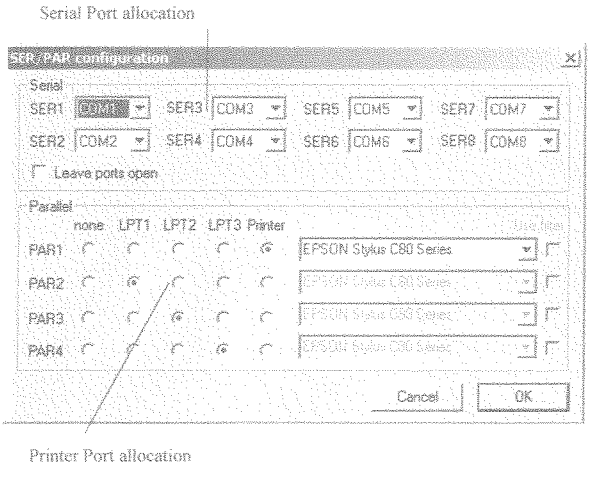
That is all there is to it. Don't forget that the other PC you put it into will not see it as WIN2_ unless you tell it to as in STEP 1 above.

You can use the option to search for a QXL.WIN file to set this up as well. The Search option can also be used to read from pre written CD drives such as those sold by Q-Celt.

The 'DOS' Window

QPC2 has had the ability to read files from the Windows section of the hard drive for some time but there were limitations on the files that could be seen because the 36 character filename limit meant some files were invisible.
The addition of the DOS device allows you to specify the sub-directory that you want to read the files from and cuts that section off the filename allowing you to read files nestled deep into the Windows PC structure. It also allows you to read files in directories which have a space in the name - something not allowed by QDOS.
If you look at the example screen shot I have allocated the following filenames
`DOS2_` is `D:\QL UPLOADS\`
and
`DOS3_` is `D:\QL DOWNLOADS\`
This allows me to transfer files directly to and from the PC disk. I can download QL files from the internet to the DOS3_ device and read them straight into QPC2 and vice versa.
Configuration of this device is by editing the strings in the appropriate windows. Just edit the filename to be the right one for the subdirectory you want as the base directory. You can go down the tree from the directory you choose but not back up it. For instance:
if you make
`DOS1_ = C:\Documents\Letters\2002`
and it contains further sub-directories
`\QL\Q Branch`
you can got into these directories but not back into 'Letters'
There are are 8 DOS drives to allocate so you should be able to choose enough to meet you needs. You can always edit them at runtime and not save the configuration so the next time you fire up QPC2 it is back to the previous settings.

SER/PAR Window



This is the window which allows you to allocate the Serial and PAR ports.
The SER1_ and SER2_ ports can be set to any of the PC-available com ports which will allow the use of the PC's internal modem (Often found on com3 or com4) for QL programs. One limitation of this is that the modems found on many modern systems are, like the printers available today, cut down units which need to access the PC's CPU and other functions to work. They may be erratic when accessed in this way.
Clicking on the 'Leave Ports Open' box will stop Windows from closing the ports prematurely but, if you do that, don't forget to close the ports manually from QPC2.
You can also allocate up to 4 different parallel ports. Some PCs have extra Parallel ports set up and some of these could have been added on as extra cards. You can assign different PAR ports in QPC2 to allow access to different printers.
By checking the 'Printer' option in the list you can assign a PAR port to spool directly to the Windows Printer device which will then handle the output and free up the QPC2 job. This does not mean that every printer will print from QPC2. The situation with printers remains that most modern units will not work with QL programs. The real use of this is to have a second QL printer set up on your PC which the QL programs can read and set that to be PAR in the configuration device.

Cancel/Save/OK

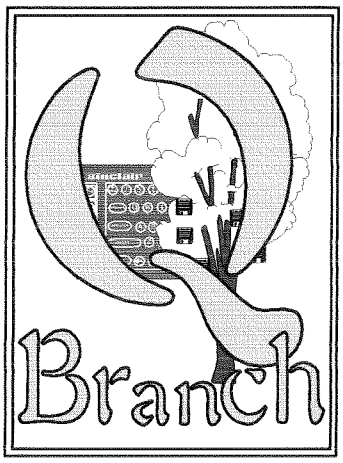
At the very bottom of the Configuration Window are the boxes marked 'CANCEL', 'SAVE' and 'OK'
'CANCEL' will quit the window but not start QPC2
'SAVE' will save the configuration settings so QPC2 will start with them the next time you start it.
'OK' will start QPC2 using these settings

Finally there is a check box which offers the option to start QPC2 directly without displaying the Configuration Window. Once you have it set how you want it check this box and you will get an instant QPC2. If you need to re-configure it just hold down the SHIFT key when you start the emulator and it will be there.

Last Word

I hope that this has helped those new to QPC2 and has maybe given some experienced users an few ideas as well. As I said I think that QPC2 is probably one of the most versatile emulators available for QL Users. It is also one of the few emulators for any system that actually goes further and gives more facilities than the original system did.

Q Branch



Text 87

£ 79.00

Typset 94 £ 29.00
Fountext 94 £ 39.00
2488 drivers £ 29.00
Epson ESC/P2 drivers £ 26.00

Text 87 is the only QDOS / SMSQ wordprocessor capable of handliing the full screen on the Aurora / QXL / QPC systems. New drivers are currently being written.

PROGRAMMING

QD 98	£ 45.00
QD + QBasic	£ 59.00
QD + Qliberator + QBasic	£ 100.00
Qliberator	£ 50.00
Master Spy v 3.3	£ 30.00
QPTR	£ 30.00
Easyptr pt 1 & 2 (together)	£ 30.00
Easyptr pt 3 (C library)	£ 14.00
QMake	£ 15.00
QMon / JMon	£ 22.00
Basic Linker	£ 19.00
DISA 3	£ 31.00
QMenu	£ 14.00

Autumnal Bargains

There are still a few ED disk available for those of you who need them. The shows are coming for the Autumn and Winter and they will sell out soon.

We also still have some QL hardware going for sale. All of this will be available at the shows in the coming months so speak up before it has all gone.

There is a complete Q 40 system on offer because a previous sale fell through. Do get in touch if you are interested.

See you at the shows.

Qubide IDE intefaces : £ 45.00

Aurora Motherboards : £ 60.00

New ED Disks

£ 7.50 per box

'Just Words' by Geoff Wicks

THESAURUS, STYLE CHECK

£ 10.00 ea / ANY 2 PROGRAMS £ 18.00 / ALL 3 PROGRAMS £ 25.00
(Includes Pointer and non-pointer driven versions)
(P.E. versions need Hot_rext, WMAN and PTR_GEN or SMSQ/E to run)
Upgrades from previous versions £ 2.50 + S.A.E. New Manuals £ 1.50

QL2PC

Convert text files from QL to PC formats and much more !

Only £ 10.00 Now with HTML support !

QL Rhymes £ 10.00
Rhyming Dictionary

Spelling Crib : PD program £ 1.50 + SAE
or Free if you buy all three programs

UTILITIES

FiFi 2	£ 18.00
QSup	£ 28.00
QSpread v 3	£ 48.00
Cueshell 2	£ 15.00
Qload / Qref	£ 15.00
Disk Mate 5	£ 16.50
QPAC 1	£ 20.00
QPAC 2	£ 40.00
QTYP 2	£ 30.00
QLQ	£ 28.00

We are currently out of stock of the SuperBasic Reference Manual
Place your order now to get one as soon as it is reprinted

The SBASIC / SuperBASIC Reference Manual

The complete definitive guide to BASIC programming in QDOS / SMSQ including three disks of PD toolkits, example procedures and an electronic index.
compiled by Rich Mellor, Franz Hermann and Peter Jaeger

Over 500 pages !
£ 40.00
+ postage

News on the SuperBasic Reference Manual Coming soon !

Tel +44 (0) 1273-386030 fax +44 (0) 1273-430501
Mobile +44 (0) 7836-745501
email qbranch@qbranch.demon.co.uk web : http://www.qbranch.demon.co.uk

Q Branch

Feeling out on a limb ?
Reach out for Q Branch.
Suppliers of Quality QDOS/SMSQ products
Hardware and Software.

New Fax Number

20 LOCKS HILL, PORTSLADE,
E. SUSSEX. BN41 2LB. UK.

ProWesS

ProWesS (now free !)	£ 1.60
DATAdesign	£ 20.00
Fontutils	£ 28.00
File Search	£ 11.00
PFlist	£ 11.00
Dilwyn's Fontpack	£ Call
LINEdesign v 2.16	£ 22.00
PWfile	£ 17.50

Paragraph

The ProWesS word processor

¶ Demo version £ 1.50 + postage
Full Registered version £ 18.00

Version 2.03 available now !

Hardware

We have a small stock of second user items. Auroras / Qubides / Gold Cards / Qplanes / superHermes etc. call us to get details of the items available. These are going fast so call soon.

QXL II	**£ 100.00**
Recycled superHermes	£ 65.00 *
Recycled Gold Card	£ 45.00 *
Recycled Aurora	See over*
Qubide	See over
Qplane	£ 25.00
Aurora cables	£ 3.00
Aurora rom adaptor	£ 3.00
'Arfa Braquet'	£ 8.00
'Son of Braquet'	£ 18.00
The 'Braquet'	£ 16.00
MC plate	£ 6.50

* when available.

Qubide upgrades to version 2.01 £ 8.00

SMSQ/E

Gold Card / Atari / QXL Version

Now Only £ 32.00

Various Atari versions : call for details

QPC 2 v3 full colour version!

Now Only £ 65.00

Upgrades from QPC 1 £ 46.00 (return master disk)
Upgrades from QPC 2 v1 £ 30.00
Upgrades from QPC 2 v2 £ 14.00
If Bought/upgraded after 15/10/01 £ 5.00

Special offer !
Get Cueshell for only £15.00 with any copy of SMSQ/E or QPC 2 upgrade !

Q Branch Programs

The Knight Safe 3	£ 35.00
upgrades from previous versions	£ 5.00
Q - Count Pointer driven home accounting	£ 25.00
Q - Route v1.08C Route finding programme	£ 25.00
The Fractal Collection	£ 35.00

Post and Packing is included with Software Prices

We can accept payment by Visa, Mastercard and Switch. We also accept cheques drawn on a UK Bank in Sterling.
You can also pay directly into our bank in Euros.
Barclays Bank Acc# 79016888 sort code :20-71-02 Name :QBranch)

A short Visit of XMenu - Part 3

Jérôme Grimbert

This time, as promised, the application window:

Application window

This kind of window is the simplest for PE, because every thing is done by the user. That's why I did not talk about them yet. As usual, you can have a lot of them, all you have to provide in the working definition is a list of them.

```
static WM_appl_t wappl[nappl+1]={&aw0,&aw1,NULL};
```

This code declares a list of two application windows. Of course, you must also have filled the field of the WM_appw structure:

- **xsize** : horizontal size, in pixels
- **ysize** : vertical size, in pixels
- **xorg** : position of the left most part, in pixels, relative to the window (WM_wwork)
- **yorg** : position of the top most part, in pixels, relative to the window (WM_wwork)
- **flag** : clear and size of shadow
- **borw** : size of the border
- **borc** : colour of the border, on one byte
- **papr** : colour of the paper, on one byte
- **pspr** : pointer to the sprite to use as cursor, NULL defaults to PE cursor.
- **draw** : drawing routine, called each time the PE wants to draw the application windows
- **hit** : hit routine, called each time the PE has an event for the application windows (such as mouse moved, keypress, click, release of click...)
- **ctrl** : control routine, useful for split sections, can be NULL if no splitted sections.
- **nxsc** : max number of x sections, best left to 0 unless you want splitted sections.
- **nysc** : max number of y sections, best left to 0 unless you want splitted sections.
- **skey** : selection key
- **spr1** : spare byte
- **spr2** : spare short
- **pwcb** : part window control block, can be NULL if no splitted sections.

- Then there are two splitted section control blocks
  - for X first
    - **xinsz**
    - **xinsp**
    - **xiciw**
    - **xicic**
    - **xiback**
    - **xiink**
    - **xiblob**
    - **xipatt**
    - **xpsac**
    - **xpsbc**
    - **xpssc**
  - then for Y
    - **yinsz**
    - **yinsp**
    - **yiciw**
    - **yicic**
    - **yiback**
    - **yiink**
    - **yiblob**
    - **yipatt**
    - **ypsac**
    - **ypsbc**
    - **ypssc**

  Just put the all mighty 0 (NULL for pointers) everywhere for the time being. We will come back later to their meaning.

- pstat : pointer to whatever you want, useful for keeping an application context with the application window.

```
static WM_appw_t aw0 = /* application sub-window */
{
wind_wid,wind_hgt,  /* xsize, ysize          size */
4,2,                /* xorg, yorg          and origin */
0,                /* flag                no clear + shadow */
0,5,0377,    /* borw, borc, papr     border width/clr, paper */
NULL,        /* *pspr                pointer to sprite */

&adraw0,                /* *draw              draw routine */
&ahit0,                 /* *hit               hit routine */
NULL,                  /* *ctrl              control routine */
0,0,                   /* nxsc, nysc         max x,y sections */
K_TAB,0,0,             /* skey, 2 spare      selection keystroke */
```

```
NULL,                          /* *pwcb            part window ctrl block */
0,0,0,0,0,0,NULL,NULL,         /*                  6 short 2 pointer zero */
CL_WRROW,CL_WBARB,CL_WBARS,/* psac, psbc, pssc arrow/bar clrs */

NULL,                          /* *pwcb            part window ctrl block */
0,0,0,0,0,0,NULL,NULL,         /*                  6 short 2 pointer zero */
CL_WRROW,CL_WBARB,CL_WBARS,/* psac, psbc, pssc arrow/bar clrs */

NULL                           /* *pstat  */
};
```

Here is a possible application window. The most important things to set are the size and origin, as well as the colours. Then comes the hard part to provide the drawing and hit routines. As usual for PE in C, it's a dedicated wrapper!

```
struct WM_action adraw0 = {JSR, wm_drwaw, ADRAW0};
struct WM_action ahit0={JSR,wm_hitaw, AHIT0};
```

I'm using the 'TAB' key to select this application window, but your mileage should vary! The remaining is just filling the structure with non-aggressive value.
One thing to be aware of, if you intend to use a custom sprite (even an animation) for an application window, is to fill the field splst of the WM_wwork structure with the list of application windows (wappl in my example). So, when using a custom sprite, the list of application windows is in pappl but also in splst.

So, so far, we just have to provide the big works: the drawing routing and the hitting routine!

The drawing routine is provided the following pieces of information:

1. a pointer to the window definition (WM_wwork *)
2. a pointer to the application window definition (WM_appw *)

The hitting routine is provided the following pieces of information:

1. a pointer to the window definition (WM_wwork *)
2. a pointer to the application window definition (WM_appw *)
3. a pointer to the window status (WM_wstat *)

Drawing routine

all iop_wblb(),iop_lblb(),... whatever you want to draw on the screen, should use the chid field of the WM_wworks structure. As usually the drawing function is called to draw (or redraw) a specific application window, there is no need to use thing like wm_swdef() or wm_swapp(). Just draw your application window, and to not bother about the outside world (including outside the application window).
At the end of the drawing function, you should return 0.

Hitting routine

First, beware! This routine is going to be called not only when there is a HIT, but also whenever there is a DO, or a key press while the cursor is on the window, or even when the cursor is moved!
So, the most important thing to do in this function is to filter the event you really want!

```
static long AHIT0(wwk,apw,wst) WM_wwork_t *wwk; WM_appw_t *apw;
WM_wstat_t *wst;
{
if ((wst->kprs==K_HIT)||(wst->kprs==K_DO))
{
...
perform some task related to your application
...
wm_wdraw(wwk); /* nasty: redraw all !!! */
}
return AHIT_NORMAL;
}
```

You should return AHIT_NORMAL, unless you want to trigger a more complex event by exiting the wm_rptr() function. The code in the example is not good code, because it ends up asking for a total redraw of the windows (including all the application windows, the loose items and so on!). It is usually better to perform only a partial drawing using wm_swdef() to set the correct size and position to a specific application window, resetting to the full window with wm_swapp(), using wm_idraw() to redraw only the updated information windows and so on.

Pan/Scroll and Splitted section

This is a classic, it uses the menu item of PE. Contrary to loose items which can be placed anywhere and could be different in shape and kind from each other, menu items must be arranged in a grid. They can style be different in shape and kind, but the grid must be ready to accomodate the biggest. In fact, the columns and rows can be of various sizes, but the width of one column cannot be changed according to the position row and vice-versa. There are columns and rows, and they cross each other to make the grid!

But this will be for another time, where we will build a simple application which uses a menu, and then we will add the possibility to split the menu. This will also gives a complete example of the various parts already seen.

In the mean time, do not hesitate to experiment!

The Q60

Dilwyn Jones

For some weeks I had a Q60 on review loan from D&D Systems. Sadly it had to go back eventually, for although I had fallen in love with it by then, I had neither the space nor the cash to buy it. This won't stop me dreaming of becoming the proud owner of one though.

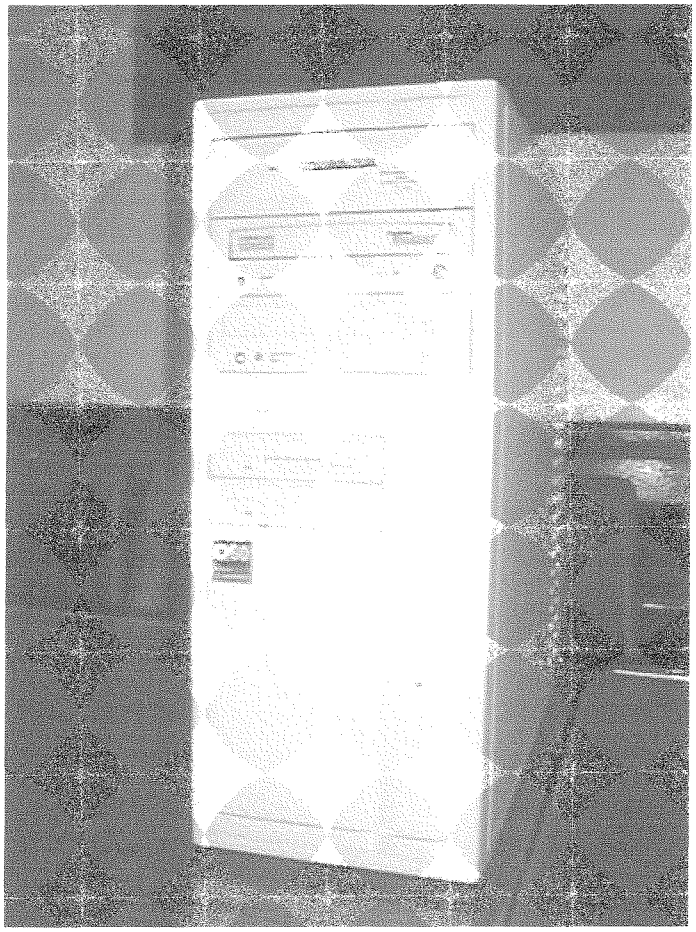
The above paragraph will make me seem biased in favour of the Q60. Yes, that's probably true. It is certainly a bit of a dream machine for QLers. For the last couple of years I've beem happily using a QPC emulator on a fairly old 333MHz PC, supplemented with a Super Gold Card and Aurora based MinisQL machine. I've always been a fan of QPC2 and when the Q60 first arrived I was wondering how the two systems would square up and how I'd make the switch from the QPC machine to the Q60.

There are some uncertainties mentioned in this review. It would have been quite easy to get in touch with Dennis Smith or Derek Stewart (the pair behind D&D Systems) to check these points, but as I am armed with their literature and manuals, leaving these questions unanswered will probably help to give a clue as to how many unanswered questions the company's literature leaves.

The Q60 initially arrived without a manual, which followed on in the post later. The Q60 was one which had been in use at D&D for a little while and was ready configured and so on. The Q60 came with a preinstalled software bundle, which according to D&D literature is an optional extra with the type 'B' build of the Q60 (more about this below) much of which will already be familiar to seasoned QLers, although I found it difficult to get to grips with so much software installed in an environment I wasn't familiar with. What I forgot to ask was whether this software is supplied on CD-ROM or what, as although my Q60 came with the software on a hard disk, the company's literature implies that built Q60 systems come with all cables etc for hard disks in place, but no mention of hard disks except on the Type B build. Now I was familiar with SMSQ/E. I was familiar with most of the programs too, but they are set up differently on my system so it took a while to get used to this, which was a little unfortunate for me as I had started a new job a short while before and try as I might I just simply couldn't make enough time to get to know the machine early on in the period it was here and so I can't really give the depth of review of the software I ought to. As the software was already on the hard disk of the Q60, I would have felt a bit more secure having a backup copy of all this software just in case something went wrong. Of course, to an experienced QLer like me it would have been quite easy to make disk backups of all the software.

The software bundle consists of several commercial programs from Jochen Merz Software, including QD, FiFi (file finder) and the ever essential QPAC1 (calculator, clock, calendar, typewriter and other useful little utility programs) and QPAC2. A huge amount of PD software with everything from sound file format converter to Prowess and the Lynx HTML viewer is included. The literature states that this bundle may well change over time, so I won't go into too much detail

about it in case it's completely changed by the time you read this. The manual states that three support disks come with the Q60, but I never saw these. The review Q60 came with a fairly generous hard disk (Type B build machines usually have a 20GB hard disk) and was fitted with a compact flash memory card reader and a CD-ROM drive. I had no memory card to test it with, but from what I've been told by others using them, these are IDE devices which means the computer sees them pretty much as a conventional drive like a hard disk or floppy disk. You can save to them, as well as loading from them and they are quite fast. Compact Flash cards are widely used in digital cameras for example, they are quite small and compact as their name implies, as well as being rugged and reliable. All sorts of memory capacities are included and seem to be the flash memory card of choice for most people as SmartMedia seems to be less used and available in smaller capacities than Compact Flash.



You can acquire a Q60 either as a main board assembly or as one of two types of cased system, the "type A" build and the "type B" build. The main board assembly consists of a fairly small (just over 8 inch by 6 inch) motherboard, memory (maximum 128MB!), VGA lead, sound adaptor, i/o card, manual, support disks and the operating system. You can get either SMSQ/E (which I guess most people will probably opt for) or QDOS Classic by Mark Swift, which is both free and has the benefit of being QDOS if you prefer sticking to a tried, trusted and very familiar operating system environment. QDOS Classic may also be preferable to those who have used the Amiga emulator. For Linux buffs, 68K Linux is also available on a CD for use with the Q60. The I/O card gives access to 2 floppy disks, 2 hard disks, 2 serial ports, 1 parallel port and a game port for a joystick.

The Type A assembly is a system built into a PC style Mini Tower case, and comes with memory, VGA lead, sound adaptor and IDC cables for the drives (but no drives included), power supply and a total of 5 drive bays in the case, 2 of which are front cover 3.5 inch bays. This should give ample room for you to set up whatever configuration of drives you want, e.g. two 3.5 inch floppy drives, a hard disk, a CD-ROM and compact flash adaptor drive. This system is perhaps best suited to the tinkerer who would like a basic system which can be built to his/her requirements. For example, a person who has access to cheap sources of drives, speakers and so on and has the experience to put all these together.

The Type B assembly comes with 20GB hard disk, 1.44 MB floppy disk, 54xCD-ROM drive and I/O card. D&D say that any additional internal equipment can be fitted for free. Both systems come with a 3-button mouse, and the Type B also comes with a neat 3 speaker system including a sub-woofer speaker and a keyboard. To save on postage costs presumably keyboards and speakers are not supplied with overseas orders as it is anticipated that local language keyboards for example should be available locally and users can purchase speakers to their own specification.

Probably the main attraction of the Q60 for QLers is the sheer speed. It does come with a 68060 processor which can be up to 80MHz. The Q60 benchmarks are certainly impressive when you have been used to an original QL. I ran some of the QL benchmarking programs on this Q60 and for some reason I couldn't match Peter's published benchmarks when I tried to test the Q60 and assume that I was making some mistake. If you'd like to

see Peter's benchmarks, just ask for a copy of the Q60 leaflet from D&D Systems.

In some respects, the benchmarks which involve testing SBASIC are not really fair because SBASIC is so different to SuperBASIC in the way it's built and used - SBASIC runs more like a compiled basic, while SuperBASIC is interpreted, so any speed comparisons are not really fair. To some extents, comparisons with the QPC2 emulator are tricky, because the speed of this emulator depends to some extent on the speed of the processor in the host PC - these tests give results obtained on a Windows NT system on a 600MHz Athlon processor. Suffice to say that as far as hardware based QL systems are concerned, anything from a standard QL to even a Q40 trails well behind the Q60!

The speed is not confined to the speed of running QL programs. It boots up pretty quick too, especially when you've been used to a PC system like mine. As the OS can be booted from ROM, it can even be faster than some QDOS based systems booting up twice in effect into SMSQ/E.

Perhaps the second reason why many of us would go for the Q60 is the high colour display. Using both 512x256 (QL resolution) and 1024x512 resolution displays, the Q60 can display 16 bit colour using 65,536 colours. Now it's fair to say that not many QL programs use this depth of colour yet, although a surprising number of programs designed for the older QL modes will actually run in this mode, thanks to the way in which the operating system supports the use of the older colour modes alongside the newer ones. The Q60 manual which came with the review computer included the documentation for SMSQ/E and for the so-called colour drivers, better known as GD2 or Graphics Device Interface version 2. Between them, both documents explain in a fairly complete no-nonsense style how to use the new colours from both SBASIC and assembler. SBASIC includes a set of new commands to handle the new colours along with extensions to existing ones such as how to change ink and paper colours for the new schemes. The new colours include such names as "faded purple", "peach", "shocking pink". Suffice to say that when you start playing with the colours you quickly get addicted if you enjoy graphics. Although it takes a while to get used to writing programs for use with the new colours, provided you study the examples given and allow yourself time, it's not that hard to write BASIC programs to use the new colours. If you have no real experience of writing BASIC programs, this would be a good incentive to study SBASIC, because you really can achieve some nice programs from BASIC on this machine. At the moment, programming tools like Easyptr don't really make use of the new colours until someone updates them, although like most other QL programs, they will run happily in the new environment in lower numbers of colours. Some QL programmers have already proved, though, that much more can be achieved with a little effort and I'm sure that given time we'll all be enjoying the benefits of 16 bit colour! Some of the programs supplied will make use of the new colours. Photon allows you to view JPEG files for example. I was shown some really impressive pictures which had been converted from other computers and to see pictures of this kind on a QL system (whether it be on a Q60 or a QPC system using 16 bit colour) does rather impress you!

QLers have never really been blessed with much in the way of sound facilities beyond the humble BEEP, apart from a few experimental sound cards way back in the early days of the QL. Well, on the Q60, you can enjoy full 20kHz stereo audio. Fire up QSPlayer (the Q60 sound player) and try out some of the sampled sounds supplied through the rather nice 3 speaker system. Not bad! The Q60 uses the _ub or Unsigned Byte sound format, a fairly basic uncompressed sound file format. Another program called SOX from Jonathan Hudson is available should you wish to try converting sounds from other file formats such as the fairly common Windows .WAV file formats to increase the number of sounds available. Although I didn't get time to test it during the time the Q60 was in my possession, a sound device driver from Simon Goodwin is also supplied so that you can access the sound facilites from BASIC. Users of QDOS CLassic on the Amiga emulator may already be familiar with the QL Sampled Sound System thanks to the work of Simon Goodwin and Mark Swift.

As far as hardware connections to the Q60 are concerned, it is very like a PC in some respects. The plugs and sockets are mostly the same ones as you'd find on older PCs. The 5 pin keyboard connector, the VGA connection for video monitors, 3.5 mm stereo socket for audio, a 15 pin socket for joystick, 9 and 25 pin sockets for serial ports, the usual PC-style parallel printer

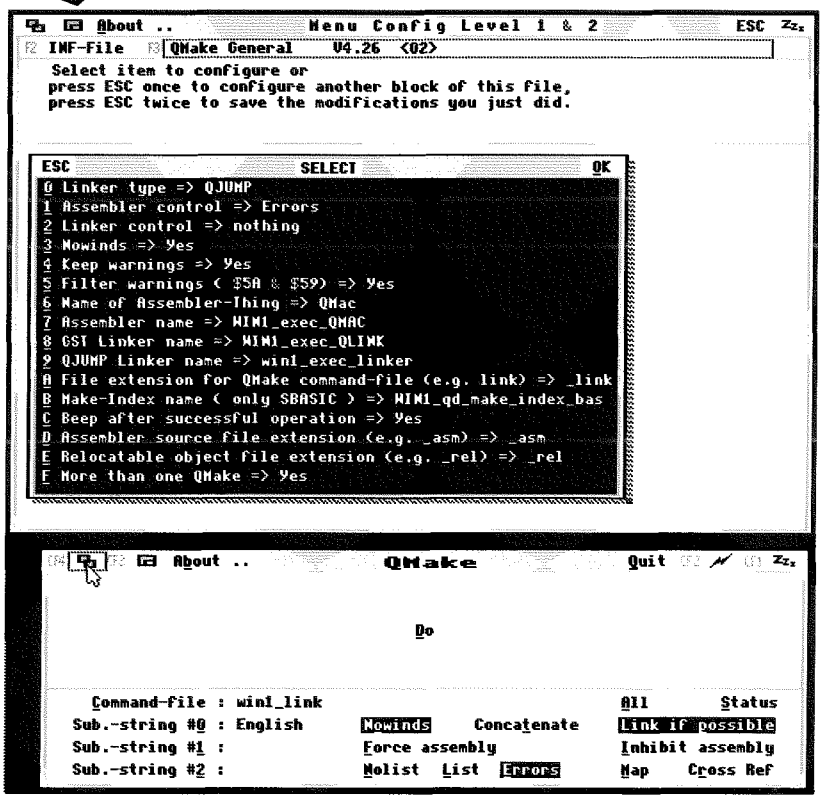
JOCHEN MERZ SOFTWARE

Im stillen Winkel 12 D-47169 Duisburg
Tel. 0203 502011 Fax 0203 502012
http://smsq.j-m-s.com

QMAKE

QMAKE, a pointer-driven MAKE program which works on LINK files suitable for QLINK and Tony Tebby's own linker, will not only allow you to link easily language-specific or machine-dependant versions, it also takes care of CCT files etc.

The snapshots show some of the configuration features and the QMAKE window when opened. QMAKE can contract to a button whilst working. Ideal for linking SMSQ/E ... Requires QMAC V1.06 or higher. Revised QMAKE manual for version 4.27..



QMAKE V4.27 **EUR 22,90**

Updated manual for V4.27 **EUR 6,-**

TERMS OF PAYMENT

Postage and package **[Germany]** EUR 4,86 (if total value of goods is up to EUR 25,- then only EUR 3,32).
[Europe] £4,50 / EUR 7,41 (if total value of goods is under £15 / EUR 25,56 then only £3 / EUR 4,86).
[Overseas] between US$7.50 (1 item) and US$17.50 (maximum).
Cheques in EURO and International Money Order in EURO (both have to be drawn on German banks).
We accept Internation Reply Coupons as payment (can bo bought at any post office): 1 IRC = EUR 1.02 redeeming value.
Exact prices are given in EUR ... the price in £ and/or US$ are approx. conversion based on the exchange rate at the time of creation of this ad. Prices may be adjusted in case the exchange rate falls considerably out of this range in both directions!
We can charge your credit cards in £'s, US$, EURO or in DM - please state the currency you prefer.
Please do not send any UK bank cheques in £ - our banks have increased the fee for handling them by 600% (no joke!) so we cannot accept them anymore, unless you add £6 for clearing the cheque.
E&OE.

port cable will do for such printers and so on. An SVGA monitor can be powered from the second socket on the back of the case, so as you switch off the Q60, the monitor will go off too, and it saves the need for a separate wall power socket for a monitor. If it comes to that, and with a suitable I/O card, you can have more than one parallel port. An example might be a business requiring a standard printer and a labelling printer. With an I/O card able to handle more than one parallel port, both printers could be connected. There are some notes in the manual which warn against some aspects of printer ports - the standard parallel port driver assumes that the parallel port is IEEE1284 compatible (ECP) and it will normally operate in SPP FIFO mode and the port can also operate in original PC mode. Other add-on cards may be useable in the ISA (Industry Standard Architecture) slots but may of course require Q60-specific drivers.

The hardware is extensively documented in the accompanying manual. The SMSQ/E version for Q60 is also well documented, giving all the Q60-specific information such as how to handle the hard disk system. It even supports the use of partitions and gives great flexibility in how you set it all up. Hard disks have to be formatted as an SMSQ drive. Both QL and MSDOS format floppy disks can be handled, which may well prove useful for the purposes of file transfer, for example. Partitioning a hard disk is achieved with the mkpart program, which is an SMSQ/E program which could be used to set up SMSQ partitions and Linux partitions for example if you wish to use your Q60 for both Linux and SMSQ/E. As I'm not a Linux buff and issues like partitioning frankly frighten me, I left well alone!

Different QL systems have different format hard disk systems. Some use the QXL.WIN container file format, others use a Qubide or Miracle hard disk format. With the advent of software such as Thierry Godefroy's ATAPI/CD drivers and software such as QCDEZE from people like Duncan Neithercutt this distinction between systems blurs, because programs like this can browse CDs in QXL.WIN format on a Q60 for example. Try one of the Q-Celt CD-ROMs for example, which are mostly in QXL.WIN format. Programs like QCDEZE make it easy to browse through those. It's high time we were able to widely use CD-ROMs for distribution of large software collections like clipart collections on QL systems. The Q60 can come with a built in CD-ROM drive, which I found quite useful, but then I'm biased - I did have a hand in creating many of the available CD-ROMs for QL systems! D&D's literature implied that it won't be long before SMSQ/E is able to write to CD-Rs as well, though at the time of writing I haven't heard any more about this.

If you have a copy of George Gwilt's FPU extensions, you can also make use of the floating pont unit for faster calculations and so on. It can really make a difference in mathematically intensive software.

So how much software is available for the Q60? Well, I originally thought not much, in fact I expected to be able to count it on the fingers of one or two hands and was pleased to be proven quite wrong! In fact when I sat down and went through everything which is either written specifically for the Q60 or just runs on a Q60 or comes in useful on a Q60, the answer is a surprising amount. I went through my extensive collection of QL software and found that surprisingly few programs would actually fail on a Q60. Of those that did, patching software and so on exists either specifically for the Q60 or for SMSQ/E in general (e.g. some software uses the MOVEP machine code instruction which doesn't exists on a 68060 processor like the one in the Q60 and causes illegal instruction errors but a simple hack exists for these types of programs).

I really had meant this to be an in depth and exhaustive review of a Q60 but that intention evaporated when I quickly realised there isn't much you can do except praise a Q60 and list its features. The review machine performed faultlessly in the weeks I had it on loan here. There did prove to be one hell of a lot to get to grips with and having just started a new job this year I found I couldn't spare the time to do justice to this machine in the level of detail I would have liked. In truth the software was set up in such a way that I could do pretty well what I wanted without anything getting too much in the way.

Given my existing love of QPC2 I had expected I might be biased against the Q60, or alternatively any new found love of the Q60 might have stunted my pride in my QPC machine, but no, I have been more than happy using both. I found myself using one computer for some types of jobs and using the other for other types of jobs. Despite the vast differences between a PC+QPC2 and a Q60 system, I really do believe that unless you have an overriding need to have a dual platform Windows and

SMSQ/E machine you will be perfectly happy with either. The Q60 is a great choice for someone who just wants the best possible QL system or a QL and Linux, especially if you dislike Windows.
By the time you read this review, D&D will have sold out the entire first production run of Q60s and planning if not already selling the next run. With a machine of this calibre and with the dedication of people like Dennis Smith and Derek Stewart and the full support of designer Peter Graf, I am 100% convinced this machine will be a certain success, it really deserves to be. I have no hesitation in recommending this computer - every QLer should have one! If I'd had enough money to hand when it came to the time to hand it back, I'd have bought this machine without hesitation.

Programming with QPTR - Part 2

Wolfgang Lenerz

Continuing on from last time's instalment, here is the new part of the series on how to use QPTR. As usual, any comments are welcome.

II - LEVEL II: Definition of the lists and sprites

If you want a window to look at least somewhat interesting, you will have to dress it up a bit - so the Level II pointers should not all be 0, but should, indeed, point to something. This is what is done by the level II functions: Level II defines the (pointer and other) sprites and sub-window lists.

A - The Sprites: "wptr"

Contrary to games computers, here a "Sprite" is just a kind of image visible on the screen, which is not "independently animated". The most typical example would of course be the mouse pointer. This is a sprite, directed over the screen by a mouse or the cursor keys. It can be an arrow, or a cross (as in FiFi) or almost anything. A sprite can also be an image that is not mobile - once it is drawn it remains where it is. The mouse pointer sprite is actually exceptional in that it can move around the screen. For example of a more normal sprite, the icon used to make a window move around the screen is, in itself, a sprite (when hit, the pointer changes to that sprite).

So, the pointer used by the application is a sprite. Each primary and secondary window can have its own sprite - as can application sub-windows. In QD, the sprite is in the shape of a cursor (blinking or not), in Disktool, it is in the shape of a disk, in FiFi it has the shape of a cross etc... You will notice that the pointer sprite "looses" its specific shape as soon as it leaves an application's primary window: as soon as you put the pointer over another application, it takes the shape given to it by that application - provided, of course, that the application has managed windows and is unlocked (of course, several applications may have the same pointer sprite). The pointer over an unmanaged and unlocked window is either an arrow or a "K", depending on whether or not the application is waiting for a keystroke. Locked windows always have another default pointer, a padlock. One cannot change these default sprites.

If each application can have its own sprite as pointer, it means that each application must define this sprite. If it doesn't (wptr=0) a pointer by default will be used, i.e. the famous little arrow.

The sprite definition is built in an area of memory which must previously have been reserved by the RESPR or ALCHP (if you have Toolkit II) functions. wptr is then simply the address of this memory area:

`wptr=`*`ALCHP`*`(size)` or `wptr=`*`RESPR`*`(size)`

Now it "only" remains to find out how much memory you should reserve (this is not a fixed amount, it varies from sprite to sprite) - and then you have to fill the memory area with the data for the sprite you wish to have.

The size of this memory area depends strictly on the size of the sprite: a small sprite will need less memory than a large sprite - which seems quite logical. For the time being, sprites are limited to 64 pixels in each direction. This may seem small, but is actually not bad.

Sprites are 'printed' to the screen in a similar way to characters, i.e. imagine a grid of columns and rows. Each element, corresponding to one pixel on the screen, can be either on or off - but here, you can not only determine whether the pixel is on or off, but also in what colour is should be 'on'.

The size of the sprite thus depends on the number of columns and rows. Suppose we want

to define a sprites in a 10 by 10 grid (10 lines with 10 rows - 10x10 pixels). To define the sprite, we read these rows and columns into an array. The array will be a normal SuperBasic string array, which, with a great leap of imagination, we shall call "sprite$" in the examples. For a 10 by 10 sprite, this array must be DIMensioned as follows:

```
DIM sprite$(9,10)
```

or more generally:

```
DIM sprite$(rows-1,columns)
```

where rows and columns are the number of lines and columns respectively. The "rows-1" is because the first dimension of a sprite is sprite$(0). Thus, by using DIM sprite$(rows-1,columns) we do get an array with the required number of lines and columns.

That still doesn't tell us what value the 'size' should be. This can be obtained with the SPRSP function (SPrite Reserve SPace), which is used as follows:

```
size= SPRSP (columns, rows)
```

where, again, rows and columns are the number of columns and lines. Note the reverse order of the parameters: columns first, rows second (this is the other way round in the DIM statement).
So, attention:

*** do not state SPRSP (rows, columns), nor SPRSP (columns, rows-1) - it's (columns, rows)!.**

*** you must double the number of columns if the sprite is a mode 8 sprite, because, indeed, each pixel is twice as large in that mode...**

Thus, to reserve sufficient memory, you should proceed as follows:

```
size= SPRSP (columns,rows):
address= RESPR (size)
```

or:

```
address= RESPR ( SPRSP(columns,rows))
```

to save on a variable (of course, RESPR can be replaced by ALCHP).

Once enough memory is reserved, the sprite needs to be defined. This is most easily obtained by using the **SPSET** (**SP**rite **SET**) command:

```
SPSET address, ori_x, ori_y, mode, sprite$
```

-› * **address** is the address obtained by the RESPR, as mentioned above;

-› * **ori_x** and **ori_y** are the x and y "origins" within the sprite. It may seem curious that a sprite has origins, as the sprite (if used as a pointer), may freely move about the screen and thus its origin changes every time. Actually, these are the origins within the sprite: A sprite can be quite large, but there must be one point as of which you consider that the sprite is inside of, say, an item or a window: this is dertermined by the origin of the sprite. Suppose you have a sprite in shape of an arrow, you may wish that the point of the arrow should be the origin of the sprite, as most people will use that to point to the vaious options... So you set the origin of the sprite to be the point of the arrow.

-› * **mode** is the colour mode in which the sprite is to be drawn: 4 or 8

-› * **sprite$** is the array we have defined above (rows-1,columns).

Of course, this array must have been filled in before using the SPSET command. This is fortunately quite easy: Each row of the array is made up as follows, using a white arrow outlined in black as an example:

```
90  DATA '   a   '
100 DATA '  awa  '
120 DATA ' awwwa '
130 DATA 'awawawa'
140 DATA '  awa  '
150 DATA '  awa  '
160 DATA '  awa  '
170 DATA '  awa  '
180 DATA '  aaa  '
```

Thus our array is filled in by a program such as follows:

```
10 RESTORE 80
20 READ rows,columns
30 DIM sprite$(rows-1,columns)
40 FOR n=0 TO rows
50   READ mydata$
60   sprite$(n)=mydata$
70 END FOR n
80 DATA 8,7  : rem the number of rows & cols
90 DATA  '   a   '
100 DATA '  awa  '
```

Independent QL Users Group

Worldwide Membership is by subscription
only, and offers the following benefits:
Monthly Newsletter - up to 40 pages
Massive Software Library - All Free !
Free Helpline and Workshops
Regional Sub-Groups. One near you?
Advice on Software and Hardware problems
Subscription just £14 for UK members
Overseas subscription £17

Barclaycard: Visa: Access: Mastercard

Now in our Nineteenth Year

Further details from the Membership Secretary

John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH
Tel. +44(0)1618 652872

or

Visit the Quanta Web Site
http://www.quanta.org.uk
Email: membership@quanta.org.uk

```
120 DATA ' awwwa '
130 DATA 'awwwwwa'
140 DATA ' awa '
150 DATA ' awa '
160 DATA ' awa '
170 DATA ' awa '
180 DATA ' aaa '
```

In line 20, the number of rows and columns is read in (the DATA in line 80). After that, the array is DIMmed and the loop reads the strings from lines 90 to 180, which are used to fill in the array. There only remains to explain the meaning of these strings:

Let's start with line 90. Each character in this string stands for ONE PIXEL. Line 90 is thus the uppermost row of the sprite. It is composed of three spaces, an 'a' and again three spaces. Each character has a special meaning: A space means that this pixel will be "transparent": it will let shine through whatever lies beneath this pixel of the sprite. An 'a' means that the pixel will be black. The letters for the other colours are:

a - black
u - blue *
r - red
m - magenta *
g - green
c - cyan *
y - yellow *
w - white
space - "transparent"

The colours marked with an asterisk ('*') can only be used for mode 8 sprites.

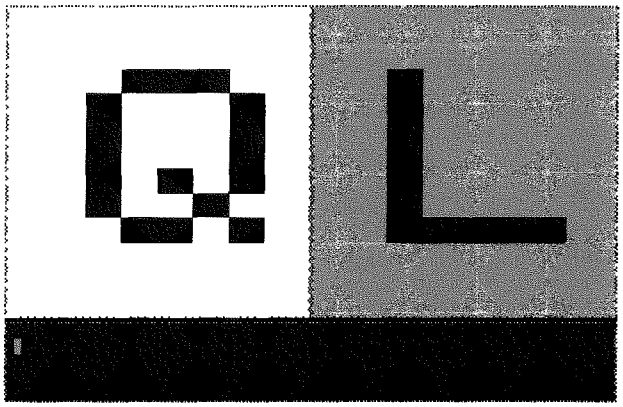
In our example, we can thus see that line 100 is composed of two transparent pixels, a black pixel, a white pixel, a black pixel and, again, several transparent pixels. In fact, the black pixels encase the white pixels. And so on for the other lines - and now we have defined the sprite. As of now, whenever we need the address of a sprite, 'wptr' will be a valid address we can use.

More next time!

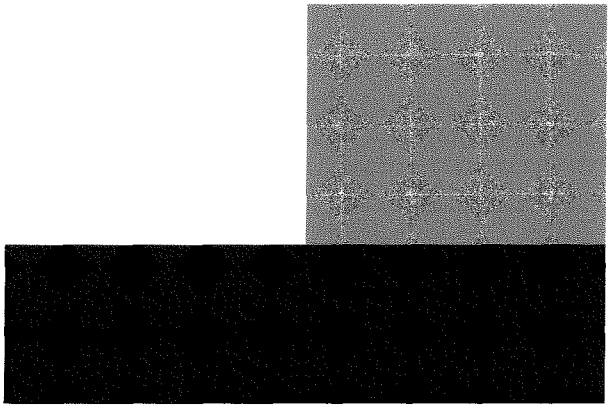
QL Logo

Dilwyn Jones

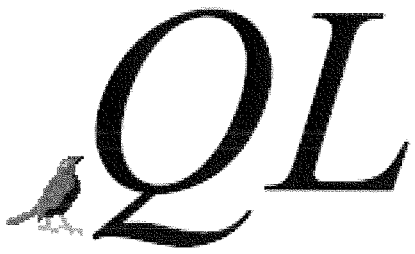
Some time ago the idea of finding a logo for the QL was floated among the QL community. Various suggestions were made and as far as I know no real consensus was arrived at. Since then, I've kept a page about this idea on my website and there has been a slow but sure contribution of ideas. Many of these might be suitable for T-shirts, mouse mats, magazine logos, anything which might help promote the QL. Some of the ideas contributed are traditional QL symbols such as the famliar red, white and black QL screen, others are much more colourful and perhaps more representative of the modern QL world.
My hope was that we could come up with something everyone would associate with the QL, in much the same way as the penguin symbol is with Linux. As far as my original idea was concerned, the best symbol of the QL is either a QL picture, or the red and white startup screen, or the letters 'QL', or the logo moulded on the original QL case! So here is my first proposal. As far as I'm concerned, anyone can use this to make a QL T-shirt or whatever - it's a GIF file of 512x256 pixel dimensions just like the startup QL screen, with the letters QL added in the chunky QL screen fount. See figure 1.



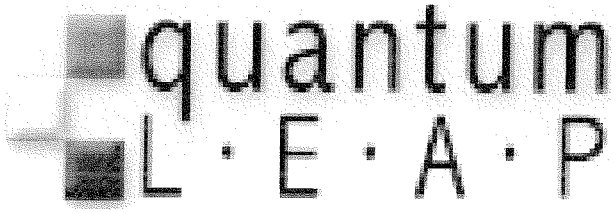
Branko Badrlijka has sent me his suggestion, a plain and simple QL monitor screen which makes for a very small graphics file which is easily resized without affecting detail.
He also suggests that a moderately thick black border may aid appearance on certain backgrounds.



Jean-Yves Rouffiac has sent me this offering for a QL logo, again a nice simple effective design. Might the little bird picture be a QuaiL perhaps???

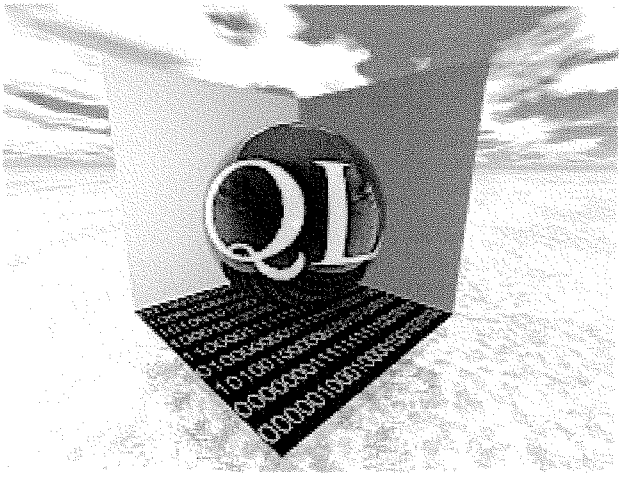


Phoebus Dokos has sent a new logo design suggestion. What do you think of this one! The blocks on the left are red (top), white (centre) and black (bottom) as per the QL monitor screen.



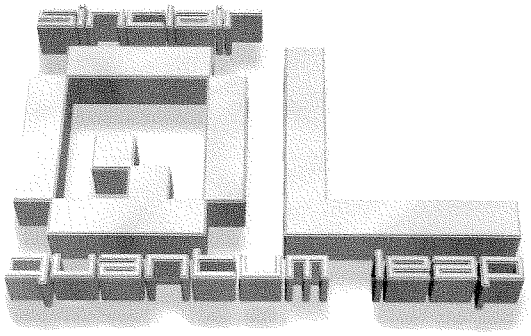
Another logo suggestion, this time from 'JH' (he says he prefers to stay anonymous!)

Another suggestion, from Nico "Acrux" Macrionitis.

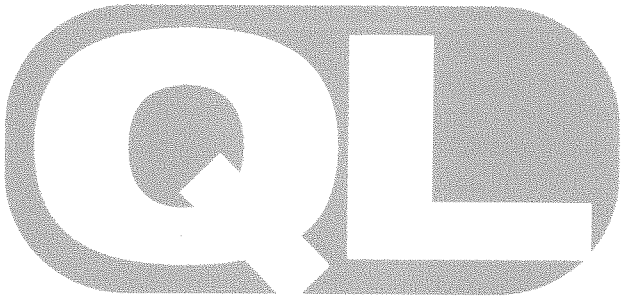


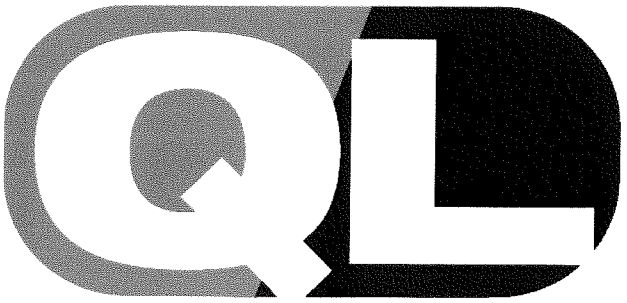
He offers the following description:
A short explanation of the adopted symbolism:
- the sides of the tetrahedron are obviously the monitor view
- the base represents the foundation of the machine language
- the black sphere is obviously the QL, a rectangle was just not as good
- the simple written QL on the sphere is in orange, like the led of ignition .

The following one comes from Melissa Ward, received on 07/07/02. Seen in colour for best results, this logo is beautifully crafted in a nice 3D golden colour.



This idea comes from Filip Dezdek in Czeck Republic (I think). A nice simple idea, well done.





JMS BBS - Final Notice

This is actually the last call to inform you to use only the faster modem line 0203 502014. I still get calls to the slower line 502013 (yes, from QL Today readers). At the moment, both lines point to the same modem anyway, but I want to free the phone number so expect it to be not working anymore from October onwards. I can't connect another modem which would answer your call just to say "don't use this number anymore, dial 502014" - so please re-program the phone number in your QTPI or whatever terminal program you use.

QPC2 HINT

Dilwyn Jones

Since I started using Windows 98 for my sins, I realised that the Taskbar options gave me a brilliantly simple to use single click launch option for QPC2, since it's the main use for my computer and made my QLing life a bit easier!

In Windows 98 and later versions, there are icons on the Taskbar for Windows Exploder, Outlook Express, View channels and Show Desktop (possibly others too in different versions). Just drag the little bar to the right of these a little further to the right to make room for an extra icon, then right-drag a copy of the QPC2 icon from the desktop to this new space on the taskbar and you can copy the QPC icon there.

Now, once you have got to the taskbar (and I always have the auto-hide option on so that the taskbar just appears when the pointer is over it) just click once on the QPC2 icon and off you go.

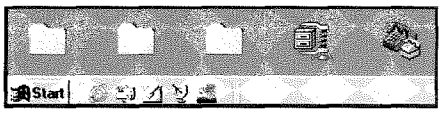
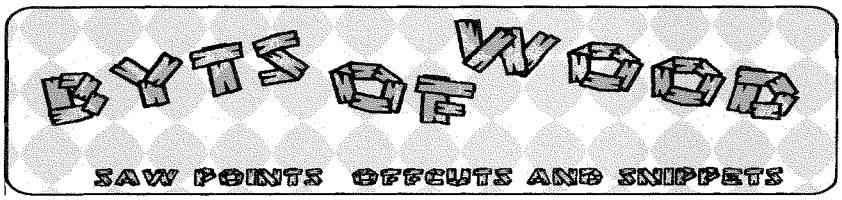


Figure 1 - the Windoze taskbar showing QPC2



As a follow-on from the comments about the past history that I started last issue's column with I noticed something in the latest round of computer trade magazines that I felt I should pass on to you all.

Black is the New off-white

There is a growing trend for manufacturers to produce black equipment. More and more of the big names in cases, keyboards and peripherals have produced black or charcoal grey units. Among the crop of units rolling off the production lines at the moment are a couple of very nice ones from AOpen. Now this company does not make cheap stuff but what it does make is usually good quality. All of its cases are usually free from the knife-sharp edges that usually make those small but incredibly painful nicks in your skin. The Power Supplies are also usually of a very good standard too. Apart from their tower cases there is one which the Qxx team should be looking at. This is a small desktop case made for the 'Flex ATX' format called the H340.

They make this in the usual off white colour but it also comes in a slinky black format with nice little chrome buttons. The 'Flex ATX' format should be fine for both the Qxx boards and for an Aurora system although it will not take a standard QL board.

The company that I work for do have a few black floppy drives and a few black CD rom drives and, together with a black mouse and keyboard, that should make for a very QL like setup.

User Revolt?

It is interesting to see that the wider computer world out there is refusing to be led by the nose in some aspects of hardware development. There is, for example, some realisation that faster and faster does not lead to a better and better computer experience. For a long while everyone who wanted to be able to hold up his head in the pub (you know the kind - Ferrari keyring for the keys to his 1987 Ford Escort) had to have the latest and fastest chip. No matter how much it was pointed out that the chip was only part of the system and that the biggest bottleneck was getting the data off the hard drive they went for it. Megahertz wars ensued between AMD and Intel and the punter suckered up to it big time.

These days that has been less of an issue. Unrealistically neither of the big chip makers seem to have noticed this and continue to crank out new models with ever more diminishing

returns in terms of the users perception. The new 2.8 GHz chip from Intel is only 400MHz faster than its previous top of the range 2.4 GHz. Now 400MHz would have been an impressive jump back in those far, far, distant days of 2000 but these days it is a mere 14% faster. Not bad for a £250 price markup, eh?

Motherboard manufacturers and system designers have, for some years, been trying to get everyone to abandon the serial & parallel ports in favour of USB and Firewire. Recently they went one further and tried to get the PS/2 port exiled to computer museumdom. ABIT brought out a board called the IT7 which boasted that it had 'no legacy devices'. In real speak this meant no parallel, serial, ISA, PS/2 or floppy drive ports. This board did not fire the imagination of the man in the street and they have been forced to bring out a new version with PS/2 and floppy interfaces on it.

Trickle Down Effect

'What has all of this to do with the QL?' you might ask. The answer is 'a lot, if we want to continue using and developing our systems over the next few years. With no serial or PS/2 ports the supply of new Serial \ PS2 Mice and keyboards will eventually dry up. Already we are forced to use an adaptor to be able to use a PS/2 style keyboard with an AT device such as superHermes and the Qxx. It will take time but these things will eventually disappear from the market with the better manufacturers discontinuing them all first.

Simon Goodwin wrote some excellent software for the Kodak DC200 camera and the QL but there are unlikely to be other cameras that use the serial port and that is how the QL communicates having only a mono directional parallel port. We really need a USB port for our systems although Nasta did seem to imply that could be a hard thing to implement.

One little benefit we will get from the updating of PC hardware is that the new serial Hard drives are beginning to emerge. Some of the newer boards have the serial interface already although the drives themselves will not be around in any quantity until early next year. (Strange how they ditch serial communication on the outside of the machine and then implement it on the inside). These new drives will have a much faster throughput than the current ones which are parallel so people who need to manipulate large amounts of

data (and on a PC a one page Word document is a large amount of data) will be inclined to start changing their systems to speed their work up. This will result in a lowering of the price of standard ATA drives and a lot of second hand ones around for us to use.

Licence to Argue

After our recent round of arguments it was also refreshing to read how much of a row has been simmering under the surface of the Microsoft users camp. Good old Bill recently changed the wording on the licence agreement for his product (you the bit that people don't read when they install the stuff - just click 'I Agree' and get on with it). It seems that he has now given himself the right to install upgrades and examine the users system without asking.

Now this may all be fodder for the paranoia merchants but you are agreeing to this stuff when you install. If you do not click that you agree then the installation aborts so you really have no choice. Some bright spark has produced a utility that allows you to continue the installation without clicking on the 'I Agree box' but what does this mean in actual fact - or in indeed in law. As someone pointed out, when we were all tearing ourselves apart labouring over tiny points in the SMSQ/E licence, no licence is really enforceable out there in the 'Wide World' but, unless the patch program does something else other than allowing you to click 'I do not Agree', just doing that small thing is just a sop to the users own ego at having 'got one over' on the great Microsoft. I suspect that just having the software installed and being used means that the licence is in force.

Nice, however, to see that the Windoze anoraks are even more fractious and petty than we are.

Text me

While I am loosely on the subject of licences I would like to point you all to something that John Hall mentioned to me at the Irish meeting. He got hold of the SMSQ/E sources and, while he also said that he did not expect to be doing anything earth shattering with them, he did say that there was one area where the files would be of great interest to many people.

Included with the files are a series of Word documents which were Tony Tebby's notes to himself while producing and refining the code. These give a lot of insight into the way that Tony thought about the process and why he did things the way that he did. These are rather long winded and, like all of Tony's writing they are, at times, a bit obtuse but between the lines there are a few interesting gems to be unearthed.

I have asked Jochen to look into maybe getting these notes summarised and put into the magazine but, if you are impatient, get a copy for yourself and have a look.

Roger Godley - a true tinkerer!

I have mentioned Roger Godley before in this column. He travels from his home in Spain each summer and spends a few weeks meeting up with other QL users. Every year he comes up with some new application that he has hacked into and changed or some hybrid piece of QL/PC hardware that he has morphed into an unusual format.

This year he had turned his attentions to Quill and Abacus and produced a new version of each of these programs which could run on higher resolution screens and actually use the screen space. He also made them so they would run multiple versions alongside each other. This means that you can have several versions of Quill all running in parallel and with job numbers so they would be easily identifiable. He has increased the number of cells available to abacus as well.

Roger Godley

RWAP SOFTWARE

QL Cash Trader v3.7 £5

A well established accounts package for the small to medium sized business, including automatic generation of profit & loss account, balance sheet, VAT returns, reports and analysis for audit trails and management decisions. Previously sold for over £100.*

QL Payroll v3.5 £5

Manage a payroll for a small to medium sized business. Handles up to 99 employees easily, producing P45s and P60s as well as the payslips on a monthly or weekly basis. Calculates tax and national insurance and is easy to update to take account of the current tax year rules.

Q-Help v1.06 £10
Q-Index v1.05 £5

Q-Help: on-screen help for SuperBASIC commands, including TK2, Turbo Toolkit, SMSQ/E and PD toolkits. Can be used to add help to your own programs - simply produce ASCII text for each help page, add an index and Q-Help automatically cross-references and displays the links.
The PD toolkits referred to are available for £2.
Q-Index: The SuperBASIC index supplied with the Reference Manual - enter a topic such as 'screen resolution' and find out the commands which relate. Launch Q-Help for further info on the chosen command.

Sidewriter v1.08 £10

Produce landscape printouts of Easel/Qspread spreadsheets and output from QL Genealogist, as well as any other standard text file. You can specify the fonts to be used on the page. Works with all EPSON compatible printers, from 9 pin dot matrix to laser printers. A most useful utility by Dilwyn Jones - you know it must be easy to use.

ProForma ESC/P2 Drivers v1.04 £8

New improved colour and monochrome printer drivers, providing up to 720dpi for all progra written for use with ProWesS, such as LineDesig Paragraph. Works on all Epson inkjet printe support binary mode compression (740, 8 models at least). 1440 dpi to follow.

QL Genealogist v3.26
Genealogy For Windo

Store your family tree f with details of their p those links build u Text files and individuals as the perfect QL versi as al Samp PC ver appear i these. QL PC version. Both program step tutorial.
** QL USERS upgrade to PC version for £25 ONLY **

s
ll of
layout.
linked to
making this
y of your family.
and QMenu as well
ale and female trees.
y since 1066 included.
- enter the details as they
nd it generates the tree from
EDCOM can be transferred to the
de to latest PC version (v5.21) for £8
easy to use and complete with a step by

D-Day MKII v3.04 £10
Grey Wolf v1.8 £8
War In The East MKII v1.24 (Upgrade Only) £5

For the gaming enthusiast - D-Day is a classic table top wargame for one or two players - you control either the Allies or the Axis forces during WWII. With the ability to define your own army set ups and a choice of 4 different scenarios, this should keep you entertained for a while.
Grey Wolf is a graphical simulation of a submarine - can you sink the enemy shipping whilst avoiding their planes and destroyers??

Image D v1.03 £10

Produce graphical representations of 3D objects - view them as wireframe, hidden line and shaded. Perspective and magnification can be controlled and views can be saved to file for subsequent printing. Multiple objects can be defined and positioned relative to each other. Simple to use yet produces excellent results.

SBASIC/SuperBASIC Reference Manual £40
Updates £6 each, £10 for 2 (Current Version - Rel 4)

Have you ever tried to write a program, but been lost as to the means of performing a certain action? This Reference Manual provides you with a full description and examples of how to use all of the keywords found on each of the different QLs, plus SMSQ/e, Toolkit II and many different public domain toolkits. Details of any possible problems are provided, together with descriptions of how to use the device drivers and how to ensure that your programs are compatible across the range of QL platforms.
This book is ideal for all QL users and is kept up to date with regular updates.
Orders are currently being taken for the next print run of this popular tome.
(Note: Price for the book does not include post & packing).

QL Cosmos v2.04 £5

Ever wondered what the stars in the sky looked like 100 years ago? Or, maybe you want to learn the constellations and names of what you see in the sky. This is the progra for you - generates pictures of the stars and planets for given place or time and provides details on these includes Halley's Comet, the Moon and the Sola planets.

Q-Route v2.00
Upgrade from v1.xx

The latest version of this popul Find the quickest route or th two places, using roads. A for this program (see els is easy and quick to u and roads to the ma

ly
lable
program
own places

Flashback (Upgrade Only) £5

The ultima mely fast and flexible, easy to the latest versions of the QL o intained. A report module is i at output in any way, including only available as an upgrade from original still available from Sector

*** SBASIC Reference Manual now Rel 4 - covers colour drivers ***
New versions of Q-Help, Q-Index and ESC/P2 ProWesS printer drivers
***** Watch out for QWord - Coming soon *****

Eden v3.08 £10
3 MKII v2.03 £8
rawn v2.01 £8
rrorday v3.1 £8
West v2.00 £5
The Lost Kingdom of Zkul v2.01 £5

A wealth of QL adventures - mainly text only.
Save the Galaxy from the ambitions of the evil dictator Nemesis.
Battle against werewolves and dracula look-alikes on a Hammer Horror set in the comical Horrorday.
Take the part of a prawn with a hangover, lost in a strange land in the hilarious Prawn.
Solve a bank-robbery by fighting the bad guys and collecting the loot in real-time old West.
Battle countless dwarves in the atmospheric Lost Kingdom of Zkul.
Return to Eden is a massive adventure over 3 disks with colourful graphics - control 3 characters in their quest to find the missing Prince.
All six adventures are available together for only £25.

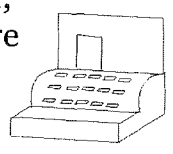
A range of games to keep both the young and the young at heart amused. Some are old favourites, like Golf and a pub quiz program (500+ questions). Others are fast, colourful arcade games. Flight simulator also now available. Plenty of variation and skill required - what more can you ask for?
All 6 programs only £28.

If v5.20 £8
II v2.07 £5
er II v2.00 £5
ne v1.2 £5
istrike v1.5 £5
ghtdeck v1.05 £10

The latest maps for Q-Route. Maps o areas of Britain have been created by them out of Big Britain Map - they will us memory and can contain more detail. A covered: Scotland, NE England, NW Englan S&W Yorkshire, Wales & Derbyshire, London area and South England.
Latest version of Q-Route is recommended.

Britain.map v1.11 £2
BIG Britain Map (needs 2MB) v2.03 £5
Various Britain Area Maps (ask for details) £2 ea.
Ireland Map v1.00 £5
Belgium Map v1.01 £2
Catalonia Map v1.03 £2

RWAP Software, 7 Common Road,
Kinsley, Pontefract, West Yorkshire
WF9 5JR
TEL: 01977 614299



Cheques in £sterling payable to 'R.Mellor'

* Also known as Trading Accounts

At the moment these programs have only got fixed sizes as defined in the code but I gave Roger a copy of the Environment Variables toolkit and suggested that he used them to allow the user to select his own screen definition. He has given the current versions to Quanta and you can get them via the librarian there I believe.
He also mentioned a project which was the reason that Roger and I got into contact in the first place - the laptop QL. Way back in 1990 Roger and I were talking about the possibility of building a truly portable QL setup. In those days laptops were pretty unwieldy things and we wanted to have a 'QL in a suitcase'. In Roger's case this was something he really needed because he lives in an area in Spain where the power supplies are rarely stable and in mine I wanted to be able to use my QL while on tour with various music groups. The PC laptop together with (at first) a QXL card and, later, the wonderful QPC2 have long since fulfilled that niche demand but Roger has persevered and tells me that he will be building this beast this winter. I will keep you informed about his progress.

QDT moving along

Just before I left for Darren Brannagh's Irish trader's 'drink up' I got a working demo copy of Jim Hunkins QDT. This is running on my desktop now. There are still a lot of 'not implemented yet' boxes but the overall impression I get is that work is progressing well.
From my brief spell of playing with the current version I can see that the program is going to be very useful in many ways. It certainly looks very good and it has to be the first program that we have had released commercially to support the new colours available to us via the SMSQ/E drivers.
This is bound to cause some differences of opinion within the QL fraternity. however, although these splits have opened before and do not, on the whole, cause us any real grief.
There is a section of QL users who feel that any use of a mouse / graphical interface / subdirectory system or any of the things which can be found on the more modern systems available today is a detraction from the original power and simplicity of the QL. It makes the system somehow 'unclean'. There are valid points, all raised and thrashed to death in many forums about the use of a mouse slows down certain applications by the need to remove the hands from the keyboard. This is not something I can easily appreciate especially in the case of the QL where the cursor and mouse pointer are often one and the same thing and can be as easily operated by the 'arrow' keys as by the mouse or trackball.
When I first started getting involved with PCs and other systems I could not get used to the fact that the mouse moved around the screen but the pointer stayed in the same place but I can now see the advantages that has to offer.
QDT is certainly something which will appeal more to the user who finds the Pointer Environment a useful addition rather than a heresy and will also, if Jim gets it right, be useful to those who find it difficult to get a boot file written when they move over to a system which has enough drive space to keep all of their files and programs.

'Drives' He Said

Believe it or not there are still many users who are not in the situation where all of their files run from hard drive. Some QPC2 users still boot their applications from floppy and I do know of one Q 40 user who still maintains the old system of booting afresh whenever he changes from one program to another and doing this via the floppy drive.
Now the QL system was designed to be multi-tasking. It was only a combination of the lack of storage space/memory and the dreadful way that the bundled PSION programs hogged all of the available RAM that prevented that from becoming the feature most regarded on the system. By the time we had a decent amount of RAM to play with the QL had been all but mothballed by AMSTRAD and we could not take advantage of the new freedom to run multiple programs to get new users.
The colour drivers for SMSQ/E should have been developed years ago but, even now, after they have been out for two years, they are very little used and remain a kind of curious toy. QDT could drive some of the users to appreciate the amount of work that has gone into getting the system to the state it is currently in.
It is not a form of heresy to embrace the newer technology that is now available to us. I have always wanted the programs that I use the most to be available as quickly and as easily as possible. This was the reason for my large button frame when I first discovered the practicality of QPAC2. Jonathan Hudson's Qascade took over from that because it was a quick elegant way to do the same job with a great deal

more flexibility. From my first use of a very early system I can see that QDT will take that task over too.

Turn the Wick Low

Darren put a lot of effort into getting this little meeting together. Given the scarcity of Irish QLers we were not expecting hordes of customers so we were not disappointed when they failed to materialise. One welcome visitor was Stuart Honeyball, a QL luminary not seen since he made his fleeting appearance at QL 2000 in Portsmouth. Unfortunately this does not presage his return to active QL duty but it was good to see him. He did show a lot of interest in the new developments in the QL though and was very amused by Darren's QL DVD show.

All in all though we spent a lot of our time there in the pub or the restaurant. Over an excellent, and reasonably priced meal, we all decided there was not enough of an opportunity for us to get together to just sit and chat so he has decided to arrange a 'Eirentoxification Weekend' next year and encourage more people to come over just for a social event. This will be something I will put into my calendar. Ireland is a great place to visit and the Wicklow Hills are a beautiful setting.

Which System?

As a final thought, after reading back through all of the above just prior to sending it off to Jochen I began to reflect on what I would say to someone who asked which what way to go to upgrade from a fairly standard QL. In truth I am not sure what I would answer. There is something very satisfying about using a system which is running the kind of operating System and software that has been designed specifically for that system. There is also something very natural about using native QL hardware. Having said that I find the QL's keyboard very annoying when I set a couple of them up to format DD disks or produce the cover disks for the magazine. (I miss the arrow key placement and the delete keys etc.)

I suppose that, if I had to make a choice I would put my main choice as QPC because it gives the user the 'everything in one box experience' A decent PC with a reasonable sized hard drive and a 17" monitor should cost no more than £500 these days and even a high spec one will roll in at under £1000 if you don't go for the very top spec. There is the issue of printing but that is one that faces every QL system and is not the fault of QPC2 or the Qxx or even SMSQ/E - it is only the way the mainstream has moved in a different direction to that set out in 1984.

I would, however, like to hear everyone's view on this and, in particular to see a poll on what systems our readers have and which ones are the main choice for them. Maybe you could let us know and we will pass this on to the general readership. I would suggest doing this by email as being easier but that would lead to a suggestion that it would bias the poll towards those main system is a PC. However you can do it any way you wish. I will start a spreadsheet for this. Keep the mails short - just three lines marked 1, 2, & 3 with each system in order of use/preference. Just QL systems or emulators of course.

You can email them to **vote@qbranch.demon.co.uk** or send them via mail to the Q Branch address with the word 'vote' in the top left hand corner (postcards preferably - even with a nice picture?) I will publish the results in the next column so get voting.

So thanks to you Darren for a nice weekend

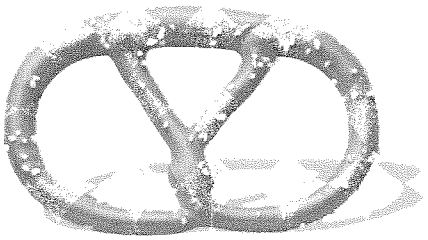
Come to Bavaria!

As we send out this issue very early, this reminder will reach most of you in time! Remember to come to the QL Show in Berchtesgaden - all details can be found in Issue 1 and 2 of Volume 7.

It is a two day event – Saturday 5th and Sunday 6th October, 2002, main day being Saturday.
Same venue as last year – Hotel Schwabenwirt, Königsseer Strasse 1, D-83471 Berchtesgaden
Tel: +49 (0)8652 2022 Fax:+49 (0) 8652 1706

Nearby is another Hotel Gruenberger (distance 50m, phone +49 8652 4560 fax ... 62254).

And remember - there's more than just a QL Show! Very nice countryside, Weizen-Beer and Pretzel, just to mention some highlights!
If you can, turn your visit into a shorter or longer holiday - we are sure you'll enjoy it!



The QL Show Agenda

QL Meeting - (NL) Eindhoven

Saturday, 2nd of November, 10:00 to 16:00
Pleincollege St. Joris, Roostenlaan 296

Quanta workshop - (GB) London

Sunday, 10th of November, 10am to 4pm
Welsh Congregational Chapel, 90 Southwark Bridge Road, London SE1

The basement area will be used for the workshop. This is small and compact, with a raised stage area leading to a recently modernised kitchen. So it will be a 'cosy' meeting!

We will occupy the stage area with tables set out with second hand QL hardware, software, books, magazines, etc, as well as some PC stuff. This will give us easy access to the kitchen.

The main area has plenty of tables and chairs available, (no need to bring your own), with a good layout to enable vistors access to the traders tables.

Electrical points are in short supply, only being two double sockets on each side. So bring plenty of extension leads and multi-way sockets. The area is relatively small, so you will not need to bring long trailing leads.

The kitchen and toilets are located in the basement area.

The Chapel is well worth a visit in itself as it is an historic piece of architectural design as well as a place of worship. It stands out on Southwark Bridge Road as a tall white faced building - late nineteenth century.

The caretaker also won an award from Southwark recently for his display of plants and climbers, etc, that adorn the front of the Chapel. It brings a lot of 'colour' to the local area. Fortunately the display was saved from being taken down by the local Council who recently completed making improvements to the pavement and road.

Parking for cars is not straighforward, as there is no car park. Yet it should be fine if you follow this guidance.

There is a small parking bay in front of the Chapel for 5 cars + bikes - although some may be occupied by other residents.

Further around Southwark Bridge Road there is a free parking section for around 9 cars - although some may be occupied (Both of these spaces are ideal for traders to drop of their stuff).

Elsewhere, Southwark have introduced a 'Permit Holders Only' parking scheme in all the local streets. Some of these apply also on Sunday. So be careful! Don't get clamped!

Nearby Great Guildford Street has Permit Holders only on Sundays - so avoid this. Yet a bit further off this street, in America Street and Wardens Place there are some spaces available at weekends.

A lot better is to go along Southwark Bridge Road past the Fire Station to Redcross Way and Lant Street where there are spaces. Also Marshalsea Road leading to Southwark Bridge Road has spaces (this is also the direct route from Borough Underground Station).

The area is very interesting for historical buildings - like the Fire Station - and small streets that surround it, etc. The Duchy Hall is also just around the corner. Also the Mint Street Park is across the road from the Chapel.

There is a petrol station, just across the road from the venue, for a fill up, and it has a shop. There are some other local shops nearby, and plenty of pubs within walking distance of the venue. Although none of them have food on Sundays - Rose and Crown, Union Street, Goldsmiths Arms, Southwark Bridge Road.

Rail - London Bridge from South East London and Kent. Also Thameslink. About 20 minutes walk to venue.

Underground - Borough, London Bridge, Monument, also Docklands Light Railway link to Borough. About five minutes walk to venue from Borough.

Buses - 43, 21

All in all it is well located for transport links.